

**RANCANG BANGUN SISTEM WARNING SEPEDA MOTOR JARAK
JAUH BERBASIS SMS (*SHORT MESSAGE SERVICE*)**

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat Mencapai Gelar Ahli Madya
Program Diploma III Ilmu Komputer



Diajukan oleh :

Frienditya Yuli Pradhifta
NIM. M.3306004

PROGRAM DIPLOMA III ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SEBELAS MARET
2009

HALAMAN PERSETUJUAN

RANCANG BANGUN SISTEM WARNING SEPEDA MOTOR JARAK JAUH BERBASIS SMS (*SORT MESSAGE SERVICE*)

Disusun Oleh

FRIENDITYA YULI PRADHIFTA
NIM. M3306004

Tugas Akhir ini telah disetujui untuk dipertahankan
Di hadapan dewan penguji
pada tanggal _____

Pembimbing Utama

Darsono, S.Si, M.Si
NIP. 19700727 199702 1 001

HALAMAN PERSEMBAHAN

Tugas Akhir ini penulis persembahkan kepada:

1. Allah SWT yang telah memberikan rahmat dan hidayahnya kepada penyusun hingga terselesaikannya tugas akhir ini.
2. Ayah, ibu, dan adik tercinta yang telah memberikan support dan do'anya kepada penulis.
3. Teman-teman teknik komputer 2006 yang telah banyak membantu, memberikan motivasi dan dukungan do'a.
4. Sahabat dekatku yang senantiasa memberikan semangat dan do'a kepada penulis.
5. **Semua pihak yang telah membantu hingga terselesainya tugas akhir ini.**

HALAMAN PENGESAHAN

RANCANG BANGUN SISTEM WARNING SEPEDA MOTOR JARAK JAUH BERBASIS SMS (*SHORT MESSAGE SERVICE*)

Disusun Oleh

FRIENDITYA YULI PRADHIFTA
NIM. M3306004

Dibimbing oleh

Darsono, S.Si, M.Si
NIP. 19700727 199702 1 001

Tugas Akhir ini telah diterima dan disahkan
oleh dewan penguji Tugas Akhir
Program Diploma III Ilmu Komputer
pada hari _____ tanggal _____

Dewan Penguji

- | | | |
|--------------|---|-----|
| 1. Penguji 1 | Darsono, S.Si, M.Si
NIP. 19700727 199702 1 001 | () |
| 2. Penguji 2 | Drs. Syamsurizal
NIP. 19561212 198803 1 001 | () |
| 3. Penguji 3 | Dra. Mania Roswitha, M.Si
NIP. 19520628 198303 2 001 | () |

Disahkan Oleh

Dekan
Fakultas MIPA UNS

Ketua Program Studi
DIII Ilmu Komputer UNS

Prof. Dr. Sutarno, M. Sc, PhD
NIP. 19600809 198612 1 001

Drs. YS. Palgunadi, M.Sc
NIP. 19560407 198303 1 004

HALAMAN MOTTO

Janganlah berlaku sombong atau berlebihan dalam menjalani hidup ini. Hidup ini penuh dengan misteri yang sulit diprediksi apa yang akan terjadi, namun janganlah takut untuk bermimpi dan berharap untuk mendapatkan kebaikan, karena dengan adanya harapan dan impian itu, hidup akan menjadi lebih berwarna dan bermakna. Raihlah harapan dan impian itu selagi waktu belum berhenti berdetak. Hadapi hidup ini dengan kerendahan hati, kesederhanaan diri, kebijaksanaan sikap, keikhlasan dan ketenangan jiwa. Semoga dengan semua itu akan menjadikan hidup ini terasa damai dan indah untuk dijalani. Segalanya berasal dari Allah SWT dan semuanya pasti akan kembali kepadaNya.

Dan kehidupan ini hanya senda gurau dan permainan, dan sesungguhnya negeri akhirat itulah kehidupan yang sebenarnya, sekiranya mereka mengetahui.

(AL- ANKABUT : 64)

KATA PENGANTAR

Puji syukur alhamdulillah penulis panjatkan kehadiran ALLAH SWT, Yang Maha Memberi Petunjuk dalam setiap kesulitan yang penulis hadapi dalam menyelesaikan laporan Tugas Akhir (TA) ini.

Pembuatan laporan ini dimaksudkan untuk memenuhi persyaratan kelulusan Program Diploma III (D3) Teknik Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Sebelas Maret Surakarta.

Berbagai upaya telah penulis lakukan untuk memperoleh kesempurnaan laporan ini. Akan tetapi, penulis menyadari bahwa masih banyak kekurangan dan kedangkalan dari laporan ini dan semoga hal ini dapat dijadikan maklum.

Ucapan terima kasih penulis sampaikan dengan tulus kepada semua pihak yang telah membantu penulis dalam penyelesaian laporan TA ini, antara lain :

6. Drs.YS. Palgunadi, M.Sc, selaku ketua Program DIII Ilmu Komputer FMIPA UNS.
7. Darsono, S.Si, M.Si, selaku pembimbing yang telah memberikan bimbingannya selama ini.
8. Teman-teman Teknik Komputer 2006, terima kasih atas kerjasama dan kekompakkannya.

Akhir kata, Penulis berharap semoga laporan ini bermanfaat bagi semua pihak pada umumnya dan bermanfaat bagi Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Sebelas Maret. Amien..

Surakarta, Juni 2009

**Frienditya Yuli
Pradhifta**

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
HALAMAN MOTTO	iv
ABSTRACT	v
ABSTRAK	vi
HALAMAN PERSEMBAHAN.....	vii
KATA PENGANTAR	viii
DAFTAR ISI	ix
DAFTAR TABEL	xi
DAFTAR GAMBAR	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah.....	1
1.2 PerumusanMasalah	
2	
1.3 Batasan Masalah	
2	
1.4 Tujuan Kegiatan	
2	
1.5 Metode Penelitian	
3	
1.6 Sistematika Penulisan 	
4	
BAB II LANDASAN TEORI.....	5

2.1 Mikrokontroller.....	5	2.1.1
Pengenalan IC Mikrokontroller AT89S51	5	
2.1.2 Organisasi Memori	7	
2.1.3 Konfigurasi Pin AT89S51.....	11	
2.1.4 Arsitektur Mikrokontroller AT89S51.....	13	
2.2 Terminal I/O HP Siemens	14	
2.3 Komunikasi Serial	16	
2.4 Perintah AT (<i>AT Command</i>)	18	
2.5 <i>Protokol Data Unit</i> (PDU)	19	
2.6 SMS Penerimaan (<i>SMS-Deliver</i>)	19	
2.7 PDU Pengirim (<i>PDU-Submit</i>)	20	
2.8 HP (<i>Handphone</i>) (<i>User, Sistem</i>)	23	
2.8.1 HP (<i>Handphone</i>) <i>User</i>	23	
2.8.2 HP (<i>Handphone</i>) <i>Sistem</i>	23	
2.9 Relay	23	
2.10 Pengenalan Bahasa Pemrograman Assembler.....	24	
2.11 Sensor <i>Reed Switch</i>	27	
2.12 Optocoupler	28	
2.13 IC LM7805	29	
2.14 IC ULN 2803	30	
BAB III DESAIN DAN PERANCANGAN.....	32	
3.1 Blok Diagram Sistem.....	32	
3.2 Diagram Alir Sistem	33	
3.3 Proses Kerja Sistem	34	
3.4 Analisis Kebutuhan	35	
3.4.1 Perangkat Keras (<i>Hardware</i>)	35	
3.4.2 Perangkat Lunak (<i>Software</i>)	42	
BAB IV HASIL DAN ANALISA	54	
4.1 Pengujian <i>Hardware</i>	54	
4.1.1 Pengujian <i>Handphone</i> Sistem.....	55	
4.1.2 Pengujian Mikrokontroller	56	

4.1.3 Interaksi Handphone dengan Mikrokontroller.....	57
4.1.4 Pemicu Relay	59
4.1.5 Komponen Sensor <i>Input</i>	60
4.2 Pengujian <i>Software</i>	63
BAB V PENUTUP	66
5.1 Kesimpulan	66
5.2 Saran	66
DAFTAR PUSTAKA	67
LAMPIRAN 1 (1 Lembar).....	68
LAMPIRAN 2 (1 Lembar).....	69
LAMPIRAN 3 (11 Lembar).....	70
LAMPIRAN 4 (1 Lembar)	81

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
HALAMAN MOTTO	iv
ABSTRACT	v
ABSTRAK	vi
HALAMAN PERSEMBAHAN.....	vii
KATA PENGANTAR	viii
DAFTAR ISI	ix
DAFTAR TABEL	xi
DAFTAR GAMBAR	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah.....	1
1.2Perumusan Masalah	
2	
1.3 Batasan Masalah	
2	
1.4 Tujuan Kegiatan	
2	
1.5 Metode Penelitian	
3	
1.6 Sistematika Penulisan 	
4	
BAB II LANDASAN TEORI.....	5

2.1 Mikrokontroller.....	5	2.1.1
Pengenalan IC Mikrokontroller AT89S51	5	
2.1.2 Organisasi Memori	7	
2.1.3 Konfigurasi Pin AT89S51.....	11	
2.1.4 Arsitektur Mikrokontroller AT89S51.....	13	
2.2 Terminal I/O HP Siemens	14	
2.3 Komunikasi Serial	16	
2.4 Perintah AT (<i>AT Command</i>)	18	
2.15 <i>Protokol Data Unit (PDU)</i>	19	
2.16 <i>SMS Penerimaan (SMS-Deliver)</i>	19	
2.17 <i>PDU Pengirim (PDU-Submit)</i>	20	
2.18 <i>HP (Handphone) (User, Sistem)</i>	23	
2.18.1 <i>HP (Handphone) User</i>	23	
2.18.2 <i>HP (Handphone) Sistem</i>	23	
2.19 <i>Relay</i>	23	
2.20 <i>Pengenalan Bahasa Pemrograman Assembler</i>	24	
2.21 <i>Sensor Reed Switch</i>	27	
2.22 <i>Optocoupler</i>	28	
2.23 <i>IC LM7805</i>	29	
2.24 <i>IC ULN 2803</i>	30	
BAB III DESAIN DAN PERANCANGAN.....	32	
3.1 Blok Diagram Sistem.....	32	
3.2 Diagram Alir Sistem	33	
3.3 Proses Kerja Sistem	34	
3.4 Analisis Kebutuhan	35	
3.4.1 Perangkat Keras (<i>Hardware</i>)	35	
3.4.2 Perangkat Lunak (<i>Software</i>)	42	
BAB IV HASIL DAN ANALISA	54	
4.1 Pengujian <i>Hardware</i>	54	
4.1.1 Pengujian <i>Handphone</i> Sistem.....	55	
4.1.2 Pengujian Mikrokontroller	56	

4.1.3 Interaksi Handphone dengan Mikrokontroller.....	57
4.1.4 Pemicu Relay	59
4.1.5 Komponen Sensor <i>Input</i>	60
4.2 Pengujian <i>Software</i>	63
BAB V PENUTUP	66
5.1 Kesimpulan	66
5.2 Saran	66
DAFTAR PUSTAKA	67
LAMPIRAN 1 (1 Lembar).....	68
LAMPIRAN 2 (1 Lembar).....	69
LAMPIRAN 3 (11 Lembar).....	70
LAMPIRAN 4 (1 Lembar)	81

DAFTAR TABEL

Tabel 2.1 <i>Serial Function Register</i>	10
Tabel 2.2 Fungsi Port 3.....	12
Tabel 2.3 Fungsi Pin pada Terminal Siemens seri 25/35/45.....	16
Tabel 2.4 Nama dan Fungsi Pin-pin DB9.....	18
Tabel 2.5 Beberapa Jenis Perintah AT.....	19
Tabel 2.6 Bagian-bagian Format SMS.....	21
Tabel 2.7 Daftar Instruksi Aritmatika Mikrokontroler.....	25
Tabel 2.8 Daftar Instruksi Logika Mikrokontroler.....	25
Tabel 2.9 Daftar Instruksi Transfer data Mikrokontroler.....	26
Tabel 2.10 Daftar Instruksi Boolean Mikrokontroler.....	27
Tabel 2.11 Daftar Instruksi lompat Mikrokontroler.....	27
Tabel 3.1 Instruksi SMS	34
Tabel 3.2 SMS peringatan.....	35
Tabel 4.1 Hasil Pengujian <i>Handphone</i> Sistem.....	55
Tabel 4.2 Tampilan Led.....	57
Tabel 4.3 Hasil Pengujian <i>Output</i> pemicu relay.....	60
Tabel 4.4 Hasil Pengujian Sensor <i>Input</i>	62
Tabel 4.5 Hasil Pengujian sistem, ketika diberi perintah SMS.....	64
Tabel 4.6 Hasil Pengujian Sistem Peringatan	64

DAFTAR GAMBAR

Gambar 2.1 Rangkaian Clock	6
Gambar 2.2 Blok Diagram Sistem	7
Gambar 2.3 Pembagian Ruang Internal Data Memori.....	8
Gambar 2.4 Konfigurasi Pin AT89S51.....	11
Gambar 2.5 Terminal Siemens Seri 25/35/45.....	15
Gambar 2.6 Konfigurasi Pin Max 232.....	17
Gambar 2.7 Konektor DB9.....	17
Gambar 2.8 Format PDU Penerima.....	20
Gambar 2.9 Format PDU Pengirim.....	20
Gambar 2.10 Gambar Fisik Relay.....	24
Gambar 2.11 <i>Reed Switch</i>	28

Gambar 2.12 Pin Optocoupler M3020.....	28
Gambar 2.13 IC LM7805.....	30
Gambar 2.14 Konfigurasi Tiap Darlington IC ULN 2803.....	31
Gambar 3.1 Blok Diagram Sistem.....	32
Gambar 3.2 Diagram Utama Alir Sistem.....	33
Gambar 3.3 Rangkaian <i>Downloader</i>	38
Gambar 3.4 Skema Rangkaian Sistem <i>warning</i> Sepeda Motor Berbasis SMS.....	39
Gambar 3.5 Rangkaian Untuk Komunikasi Serial Menggunakan IC Max 232....	40
Gambar 3.6 Skema Rangkaian Keluaran dan relay.....	41
Gambar 4.1 Sistem Kerja Rangkaian <i>warning</i> Sepeda Motor Berbasis SMS.....	54
Gambar 4.2 Program Untuk Menyalakan Led.....	56

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Tingkat kriminalitas di negara ini semakin tinggi, khususnya angka kriminalitas pencurian sepeda motor. Salah satu faktor penyebab tingginya tingkat pencurian sepeda motor ini adalah kurang efektifnya sistem *security* atau keamanan pada kendaraan bermotor sekarang ini. Dari hasil peninjauan pada bengkel – bengkel yang memasang sistem alarm sepeda motor maupun hasil survei responden pengguna sistem alarm sepeda motor diketahui bahwa sistem alarm yang ada pada kendaraan sepeda motor (sekarang ini) hanya memiliki fasilitas kunci keamanan biasa.

Berikut hasil survei responden pengguna sistem alarm sepeda motor : Sistem alarm sepeda motor dengan menggunakan saklar kombinasi. Saklar kombinasi menyebabkan mesin kendaraan tidak dapat dihidupkan bila kombinasi saklar yang dilakukan tidak benar (dapat merusak sistem *ignition* sepeda motor). Saklar – saklar ini biasanya letaknya tersembunyi, namun biasanya pencuri dapat mengetahuinya, dengan menelusuri perkabelan dari sistem *ignition* sepeda motor.

Sistem alarm sepeda motor yang dilengkapi dengan *shock* sensor. Sensor ini berfungsi untuk mendeteksi getaran yang terjadi pada kendaraan bermotor. Akan tetapi seringkali *shock* sensor mendeteksi adanya getaran pada sepeda motor (bukan karena akan dicuri), tetapi karena tertimpa sesuatu atau memang sengaja dikerjai dengan cara menggoyang-goyangkan *body* sepeda motor. Keadaan ini tentu akan memicu sistem alarm untuk membunyikan sirine atau alarm.

Sistem alarm sepeda motor dengan menggunakan sistem *remote* RF (*Radio Frekuensi*), dimana sistem alarm *security* sepeda motor ini lebih canggih dari sistem saklar kombinasi yaitu mampu melakukan sistem *lock* maupun *unlock* mesin sepeda motor dari jarak yang cukup jauh maupun dekat. Namun menu – menu atau macam *feature* yang digunakan untuk sistem alarm sepeda motor hanya terbatas atau masih kurang *secure* dan juga apabila jaraknya terlalu jauh maka sistem RF tidak dapat menjangkaunya.

Melihat dari beberapa fakta yang ada di atas dan masih tingginya tingkat pencurian kendaraan bermotor, dirasakan perlu adanya perbaikan terhadap sistem alarm yang telah ada sekarang ini. Sehingga untuk memperbaiki kekurangan yang ada dalam sistem keamanan atau *security* kendaraan bermotor khususnya roda dua (sepeda motor) maka dalam tugas akhir ini akan dibicarakan lebih lanjut.

1.2 Perumusan Masalah

Berdasarkan latar belakang masalah, maka dapat diambil rumusan masalah sebagai berikut ”Bagaimana cara rancang bangun sistem *warning* sepeda motor jarak jauh berbasis SMS (*Sort Message Service*)?”

1.3 Batasan Masalah

Penelitian ini dibatasi pada perencanaan, pembuatan dan pengujian

suatu sistem *warning* sepeda motor dengan menggunakan teknologi SMS. Sistem ini menggunakan mikrokontroler AT89S51 yang bertugas mengatur seluruh kegiatan sistem yang dirakit, yaitu mengolah inputan dari sensor ataupun SMS, mengaktifkan atau menonaktifkan relay, membangun komunikasi antara *handphone* sistem dengan *handphone user*. *Handphone* sistem yang digunakan adalah Siemens C35i. Bahasa pemrograman yang digunakan adalah bahasa pemrograman Assembler.

1.4 Tujuan

Tujuan dari tugas akhir ini adalah melakukan rancang bangun sistem pengaman sepeda motor jarak jauh dengan memanfaatkan teknologi SMS (*Short Message Service*) serta menggunakan mikrokontroller AT89S51.

1.5 Metode Penelitian

- a. Studi Literatur
 - 2.24.1 Mempelajari mikrokontroler seri AT89S51
 - 2.24.2 Mempelajari IC MAX232 dan IC ULN 2803
 - 2.24.3 Mencari referensi tentang *wiring* / jalur kelistrikan sepeda motor
 - 2.24.4 Mempelajari karakteristik SMS dengan bahasa PDU (*Protocol Data Unit*)
- b. Perencanaan dan pemrograman
 - i) Desain elektronika
 - ii) Desain mikrokontroller
 - iii) Desain algoritma / *flowchart*
 - iv) Merencanakan dan membuat peralatan dan sistem yang dibutuhkan secara *hardware* dan *software*.
- c. Pengujian alat dan analisa sistem
 - i) Pengujian *hardware* dan *software*
 - ii) Analisa *hardware* dan *software*
 - iii) Pengujian dan analisa keseluruhan sistem
- d. Kesimpulan
Kesimpulan dari hasil pembuatan alat.

1.6 Sistematika Penulisan

Sistematika pembahasan tugas akhir ini adalah sebagai berikut:

A. BAB I PENDAHULUAN

Bab ini berisi tentang latar belakang masalah, perumusan masalah, tujuan, batasan masalah, metodologi dan sistematika pembahasan yang digunakan dalam pembuatan tugas akhir ini.

B. BAB II DASAR TEORI

Bab ini berisi teori–teori pembahasan secara garis besar mengenai mikrokontroler AT89S51, Terminal I/O Telepon Seluler Siemens Seri 25/35/45, pengenalan pemrograman assembler, format PDU (*Protokol Data Unit*), Perintah AT (*AT Command*), dan komponen-komponen elektronik pendukung sistem.

C. BAB III DESAIN DAN PERANCANGAN

Bab ini membahas perencanaan perangkat keras dan perangkat lunak yang diawali dengan penjelasan diagram blok dari perencanaan secara keseluruhan, kemudian diikuti dengan penjelasan dari masing-masing bagian.

D. BAB V HASIL DAN PEMBAHASAN

Bab ini berisi tentang penjelasan setiap rangkaian penyusun alat dan hasil pengujian terhadap alat yang telah dibuat untuk mengetahui kinerja dari alat.

E. BAB IV PENUTUP

Bab ini berisi tentang kesimpulan dari hasil keseluruhan pembuatan alat dan saran untuk memperbaiki alat.

F. DAFTAR PUSTAKA

G. LAMPIRAN

BAB II

LANDASAN TEORI

2.1 Mikrokontroller

Mikrokontroller adalah sebuah chip IC yang berfungsi sebagai pengontrol rangkaian elektronik dan umumnya dapat menyimpan program di dalamnya. Ada perbedaan penting antara mikroprosesor dan mikrokontroller. Mikroprosesor merupakan CPU (*Central Processing Unit*) tanpa memori dan *input/output* pendukung sebuah komputer, sedangkan mikrokontroller umumnya terdiri dari CPU, memori I/O tertentu dan unit pendukung seperti ADC (*Analog to Digital Converter*) yang sudah terintegrasi di dalamnya. Kelebihan utama mikrokontroller adalah tersedianya RAM *internal* dan peralatan I/O pendukung sehingga ukuran *board* mikrokontroller menjadi lebih ringkas.

2.1.1 Pengenalan IC Mikrokontroller AT89S51

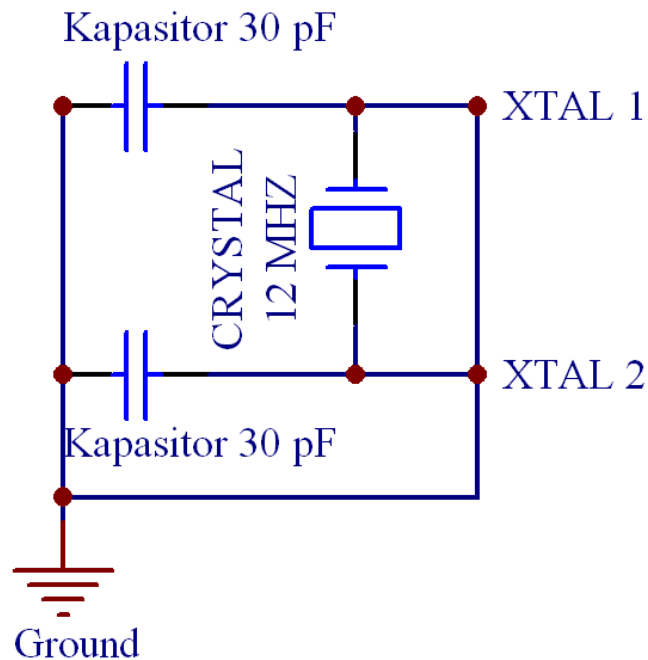
Mikrokontroller AT89S51 adalah mikrokomputer C-MOS 8 bit dengan 4 *kbytes flash memory*. Mikrokontroller ini bekerja pada daya rendah, tetapi memiliki kemampuan kerja yang cukup handal. Mikrokontroller ini diproduksi oleh ATMEL dengan teknologi memori *non volatile* artinya data tetap tersimpan jika catu daya dimatikan. Mikrokontroller ini juga kompatibel dengan mikrokontroller MCS-51 dan standart industri 80C51. Mikrokontroller ini memiliki fitur-fitur sebagai berikut :

9. Kompatibel dengan produk MCS-51.
10. 4 *kbytes* ISP *flash memory* (dapat diprogram dan dihapus sebanyak 1000 kali).
11. Tegangan operasi 4,0 volt sampai dengan 5,5 volt.
12. Frekuensi kerja : 0 Hz sampai dengan 33 Hz.
13. Tingkat program memory *lock*.
14. 128 x 8 bit RAM Internal.
15. 32 *Input/ Output* yang dapat diprogram.
16. 2 buah *timer/ counter* 16-bit.
17. 2 sumber *interrupt*.
18. *Full Duplex* UART kanal serial.
19. Bekerja pada daya rendah dan *power down modes*. *Interrupt recovery* dari *mode power down*.
20. *Watch dog timer*.
21. Data pointer ganda.
22. *Koast programming time*.
23. Pemrograman ISP yang fleksibel (*Byte* and *page mode*).

Oscillator Clock

Agar dapat mengeksekusi program, mikrokontroller membutuhkan

pulsa clock. Pulsa ini dapat dihasilkan dengan memasang rangkaian resonator pada XTAL1 dan XTAL2. Frekuensi kerja maksimum 89S51 adalah 33 MHz. Rangkaian clock/ Resonator diperlihatkan oleh Gambar 2.1. dan dihubungkan dengan kapasitor bernilai 30pF.



Gambar 2.1. Rangkaian Clock / Resonator.

Waktu eksekusi sebuah instruksi oleh mikrokontroller tergantung dari jenis instruksi dan frekuensi clock yang digunakan. Setiap instruksi memiliki panjang *byte* dan jumlah siklus yang berbeda. *Byte* instruksi (*Byte*) menandakan jumlah lokasi memori yang dipakai. Siklus instruksi menandakan jumlah *machine cycle* yang dibutuhkan. Waktu eksekusi dapat dirumuskan sebagai berikut :

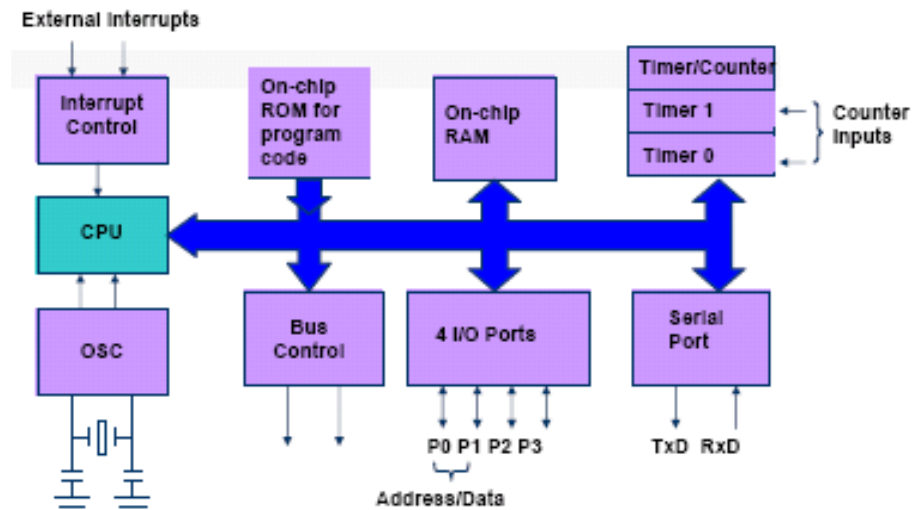
$$T_{inst} = \frac{C \times 12}{\text{frekuensi crystal}}$$

keterangan :

T_{inst} : Waktu yang dibutuhkan untuk mengeksekusi 1 instruksi (Second).

C : Jumlah *machine cycle*

2.1.2 Organisasi Memori

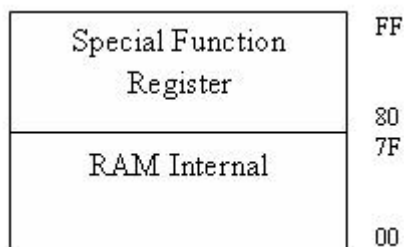


Gambar 2.2. Blok Diagram Sistem.

Memori

Pada dasarnya dalam mikrokontroller ada dua tipe memori, memori data dan memori program. Pembagian tersebut bertujuan untuk mempercepat proses kerja mikrokontroller.

Memori data berfungsi untuk menyimpan data. Memori data yang dimiliki keluarga MCS-51 sebesar 128 bytes ditambah dengan SFR sehingga jumlahnya mencapai 256 bytes. Memori program berfungsi untuk menyimpan kode program *user* yang akan dijalankan yang memiliki beberapa alamat khusus yang ditujukan untuk *reset address* (alamat yang pertama kali dituju pada saat mikrokontroller bekerja) dan *interrupt vektor address*.



Gambar 2.3. Pembagian ruang *internal* memori data.

Interrupt vektor address adalah suatu fasilitas pada mikrokontroller yang berfungsi untuk menghentikan atau menjalankan program yang lain (*program interrupt*) sampai selesai.

SFR (*Special Function Register*)

SFR adalah alamat pada memori internal yang memiliki fungsi khusus. Dalam setiap SFR memiliki fungsi dan cara pemakaian yang berbeda-beda. Berikut ini penjelasan singkat tentang SFR pada AT89S51.

- a. **Accumulator (ACC)**, digunakan sebagai register utama dalam proses aritmatika dan penyimpanan data sementara. ACC akan menyimpan hasil perkalian 8 bit terbawah dan hasil bagi. Dalam instruksi pemrograman dituliskan dengan A.
- b. **Register B**, digunakan untuk menyimpan hasil perkalian 8 bit teratas dan sisa pembagian.
- c. **Stack Pointer (SP)**, adalah register yang menunjukkan alamat dari stack teratas disimpan. Pada operasi PUSH, LCALL, proses *interrupt* dan sejenisnya akan menambah nilai pada SP. Sedangkan instruksi POP, RET, RETI dan sejenisnya akan mengurangi nilai pada SP.

- d. **Data Pointer (DPTR)**, merupakan register 16 bit yang digunakan sebagai penyimpan alamat data. Terdiri dari 8 bit DPH sebagai penyimpan *high byte* dan 8 bit DPL sebagai penyimpan *low byte*. DPTR umumnya digunakan untuk mengakses alamat pada memori eksternal.
- e. **Port 0, Port 1, Port 2, Port 3**. Port-port ini merupakan *latches* yang digunakan untuk menyimpan data yang akan ditulis dari/ke masing-masing port.
- f. **Serial Data Buffer (SBUF)**, terdiri dari 2 register yang terpisah, yaitu register penyangga pengirim (*transmit buffer*) dan penyangga penerima (*receive buffer*). Pada saat data disalin ke SBUF maka data yang sesungguhnya dikirim ke penyangga pengirim sedangkan saat data disalin dari SBUF maka sebenarnya data tersebut berasal dari penyangga penerima.
- g. **Control Register**, Register-register IP, IE, TMOD, SCON, TCON dan PCON berisi bit-bit kontrol dan status untuk sistem interupsi, *timer*, *counter*, dan port serial.
- h. **Timer Registers**, merupakan register yang digunakan untuk mengatur operasi *timer*. Register ini meliputi *Timer 1 High byte* (TH1), *Timer 0 High byte* (TH0), *Timer 1 Low byte* (TL0), *Timer 1 Low byte* (TL1), *Timer Mode* (TMOD) dan *Timer Control* (TCON).
- i. **Interrupt register**, digunakan untuk mengatur proses interrupt. Register ini meliputi *Interrupt Enable* (IE) dan *Interrupt Priority* (IP). Beberapa parameter interrupt juga terdapat pada TCON dan SCON.

Tabel 2.1 Serial Function Register.

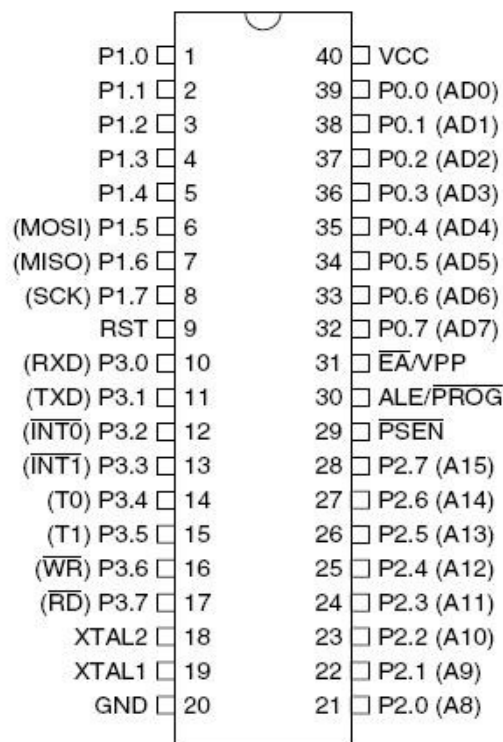
SFR	Fungsi	Alamat (Hex)
DPH	<i>Addressing external memory</i>	83
DPL	<i>Addressing enable control</i>	82
IE	<i>Interrupt enable control</i>	0A8
IP	<i>Interrupt priority</i>	0B8
PO	<i>Input-output port latch</i>	80
P1	<i>Input-output port latch</i>	90
P2	<i>Input-output port latch</i>	0A0
P3	<i>Input-output port latch</i>	0B0
PCON	<i>Program control</i>	87
SCON	<i>Serial port control</i>	98
SBUF	<i>Serial port data buffer</i>	99
TMOD	<i>Timer/ counter mode control</i>	89
TCON	<i>Timer/ counter control</i>	88
TL0	<i>Timer 0 low byte</i>	8A
TH0	<i>Timer 0 high byte</i>	8C

TL1	<i>Timer 1 low byte</i>	8B
TH1	<i>Timer 1 high byte</i>	8D

(Dyah E Indriyani, 2008)

2.1.3 Konfigurasi Pin AT89S51

Mikrokontroller AT89S51 mempunyai 40 pin dan dikemas dalam *Dual Inline Package*. Berikut ini fungsi dari setiap masing – masing pin:



Gambar 2.4 Konfigurasi Pin AT89S51.

a) Pin 1-8

Port I, merupakan *bidirectional I/O port*. Setiap keluaran tiap pin dapat dijadikan sebagai input bagi TTL. Port ini Juga dapat dikonfigurasi sebagai *multiplex low address/ data bus* selama akses ke *eksternal program* dan data memori. Port ini juga dapat menerima *code byte* selama *flash programming* dan *output* kode selama verifikasi program.

b) Pin 9

Pin Reset, sebagai masukan sinyal *reset*, yaitu kondisi *high* pada pin ini selama dua siklus mesin ketika *oscillator running reset* pada peralatan.

c) Pin 10 – 17

Port bidirectional I/O dengan *internal pull up*, dapat juga dikonfigurasi menjadi *multiplexed low address/ data bus* selama akses ke *eksternal program* dan memori data. Selain itu dapat menerima *code byte* selama *flash programming* dan *output code* selama verifikasi program. Disamping itu memiliki fungsi berkaitan dengan pengiriman data secara serial, sinyal control dan *eksternal data memory, timer*.

Tabel 2.2 Fungsi Port 3

Port/Pin	Fungsi
P3.0/10	RXD masukan port serial
P3.1/11	TXD keluaran port serial
P3.2/12	INT0 masukan interupsi 0
P3.3/13	INT1 masukan interupsi 1
P3.4/14	T0 masukan Timer/Counter 0
P3.5/15	T1 masukan Timer/Counter 1
P3.6/16	WR pulsa penulisan data memori luar
P3.7/17	RD pulsa pembacaan data memori luar

d) Pin 18 – 19

XTAL1 dan XTAL2, sebagai masukan *clock*. Pin 18 dan 19 ini digunakan jika menggunakan *clock internal*. Jika menggunakan rangkaian *clock eksternal* maka hanya pin 19 yang digunakan sedangkan pin 18 tidak digunakan/ digroundkan.

e) Pin 20

Ground

f) Pin 21 – 28

8 bit saluran I/O dua arah. *Port* ini juga digunakan untuk pengalamatan 16 bit alamat tinggi (dengan instruksi *movx @dptr*).

g) Pin 29

PSEN (*Program Store enable*), merupakan sinyal baca untuk mengeksekusi memori *eksternal*.

h) Pin 30

ALE (*Address latch enable*), merupakan pulsa yang berfungsi untuk menahan alamat rendah (A0-A7) pada *port 0* selama dilakukan proses baca/ tulis memori *eksternal*, juga berfungsi sebagai masukan pulsa program selama dilakukan pemrograman pada EEPROM *internal*.

i) Pin 31

EA (*Eksternal Access*), merupakan pin yang mengatur memori yang digunakan jika EA dihubungkan ke Vcc maka memori program yang digunakan adalah memori *internal*, sedangkan jika EA dihubungkan ke *ground* maka memori yang akan digunakan adalah memori *eksternal*.

j) Pin 32 – 39

Port 0, selain sebagai saluran I/O 8 bit dua arah juga sebagai pengalamatan 16 bit alamat

low (A0-A7) yang *demultipleks* dengan saluran bus data (D0-D7) yang digunakan pada saat mengakses ROM dan RAM eksternal.

- k) Pin 40
Vcc, masukan catu daya 5 Volt.

2.1.4 Arsitektur Mikrokontroler AT89S51

Adapun arsitektur mikrokontroler AT89S51 ini dikombinasikan dengan sebuah *Central Processing Unit* (CPU) dua jenis memori (ROM dan RAM), *input/ output*, *status mode*, dan register. Setiap bagian dihubungkan dengan jalur data bus 8 bit. Jalur data ini juga digunakan untuk berhubungan dengan sisi luar mikrokontroler melalui *port I/O* ketika ekspansi memori atau instruksi I/O.

CPU yang merupakan otak dari mikrokontroler terdiri dari :

- 3 ALU (*Aritmatic Logic Unit*)
Digunakan pada pengoperasian aritmatika dan logika. Fungsi aritmatika dasar yang dapat dibuat adalah penjumlahan, perkalian, pembagian. Sedangkan fungsi logika yang dapat dioperasikan adalah AND, OR, dan XOR, serta pergeseran *bit*, *komplemen*, *set*, *clear*, dan sebagainya.
- 4 Accumulator dan PSW (*Program Status Word*)
Accumulator (register A) merupakan register 8 bit yang berfungsi sebagai tempat tujuan dari hasil instruksi aritmatika. Akumulator juga merupakan asal dan tujuan dari instruksi logika dan sejumlah instruksi pemindahan data khusus. Banyak instruksi yang mempunyai efek tertentu, misal status tanda (positif atau negatif, nol, paritas, dan alin sebagainya) yang dikelompokkan secara bersama ke dalam sebuah register PSW.
- 5 Register SP (*Stack Pointer*)
Register 8 bit yang mempunyai fungsi khusus adalah register B yang akan melayani proses eksekusi instruksi perkalian dan penjumlahan. Register ini digunakan bersamaan dengan *accumulator*.
Register SP merupakan register *stack* (penyimpan sementara) 8 bit yang akan menyimpan *byte* terakhir yang di *push*. SP akan bertambah satu jika ada instruksi *push* dan berkurang satu jika ada instruksi *pop*.

Data sheet Mikrokontroler AT89S51 terdapat pada Lampiran 1 halaman 70.

(<http://www.atmel.com/>)

2.2 Terminal I/O HP Siemens

Telepon seluler pada umumnya menyediakan terminal *input output* (I/O) khusus yang digunakan untuk keperluan umum. Fungsi terminal I/O ini biasanya digunakan untuk pengisian baterai, terminal *handsfree*, dan komunikasi data. Komunikasi data pada telepon selular, khususnya Siemens seri 25/35/45 bekerja secara serial dengan *baud rate* 19200bps.



Gambar 2.5. Terminal Siemens seri 25/35/45.
(www.delta-electronic.com)

Terminal Siemens seri 25/35/45 tersusun atas 12 pin I/O yang mempunyai fungsi di tiap pin-nya terlihat pada Tabel 2.3.

Tabel 2.3 Fungsi pin pada terminal Siemens seri 25/35/45.

PIN	NAMA	FUNGSI	In/Out
1	GND	Ground	
2	SELF-SERVICE	Kendali pengisi baterai	In/Out
3	LOAD	Beban tegangan	In
4	BATTERY	Baterai	Out
5	DATA OUT	Data keluaran (TX)	Out
6	DATA IN	Data masukan (RX)	In
7	Z_CLK	Kendali asesoris	
8	Z_DATA	Kendali asesoris	
9	MICG	Ground untuk mic	
10	MIC	Masukan mic	In
11	AUD	Keluaran Audio	
12	AUDG	Ground untuk Audio	Out

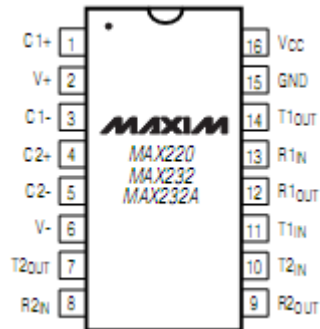
2.3 Komunikasi Serial

a. RS-232

Device pada komunikasi serial port dibagi menjadi 2 (dua) kelompok yaitu *Data Communication Equipment* (DCE) dan *Data Terminal Equipment* (DTE). Contoh dari DCE ialah *modem, plotter, scanner* dan lain-lain sedangkan contoh dari DTE ialah terminal di komputer. Spesifikasi elektronik dari serial port merujuk pada *Electronic Industry Association* (EIA) :

1. “Space” (logika 0) ialah tegangan antara +3 hingga +25 V.
2. “Mark” (logika 1) ialah tegangan antara –3 hingga –25 V.
3. Daerah antara +3V hingga –3V tidak didefinisikan /tidak terpakai
4. Tegangan *open circuit* tidak boleh melebihi 25 V.
5. Arus hubungan singkat tidak boleh melebihi 500mA.

Jika peralatan yang digunakan menggunakan logika TTL, maka sinyal serial port harus dikonversikan dahulu ke pulsa TTL sebelum digunakan, dan sebaliknya sinyal dari peralatan harus dikonversikan ke logika RS-232 sebelum diinputkan ke serial port. Konverter yang paling mudah digunakan adalah MAX-232. Di dalam IC ini terdapat *Charge Pump* yang akan membangkitkan +10 Volt dan -10 Volt dari sumber +5 Volt tunggal. Dalam IC *Dual In-line Package* (DIP) 16 pin (8 pin x 2 baris) ini terdapat 2 buah *transmitter* dan 2 *receiver* (www.toko-elektronika.com/tutorial/pcinterfacing.htm, 2009).

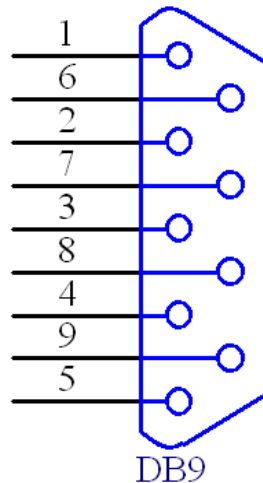


Gambar 2.6 Konfigurasi pin MAX232.

b. Konektor DB9

Konektor port serial terdiri dari 2 jenis, yaitu konektor 25 pin (DB25) dan 9 pin (DB9) yang berpasangan (*male/female*). Bentuk dari konektor DB-25 sama persis dengan port paralel. Umumnya COM1 berada di alamat 3F8H, sedangkan COM2 di alamat 2F8H (www.toko-elektronika.com/tutorial/pcinterfacing.htm, 2009).

Konfigurasi pin untuk konektor DB9 yang dipakai adalah sebagai berikut:



Gambar 2.7 Konektor DB 9.
Tabel 2.4 Nama dan fungsi pin-pin DB9.

Pin	Nama	Fungsi
1	<i>Data Carrier Detect (DCD)</i>	Sebagai pendeteksi sinyal pembawa
2	<i>Receive Data (RD)</i>	Sebagai penerima data dari DCE
3	<i>Transmit Data (TD)</i>	Sebagai pengirim data DCE
4	<i>Data Terminal Ready (DTR)</i>	Sebagai pemberi sinyal ke DCE ketika DTE siap bertindak
5	<i>Signal Ground (SG)</i>	Referensi pertanahan untuk semuanya sinyal
6	<i>Data Set Ready (DSR)</i>	Sebagai penerima sinyal ketika DCE siap bertindak
7	<i>Request To Send (RTS)</i>	Memberikan sinyal ke DCE ketika DTE mengirim data
8	<i>Clear To Send (CTS)</i>	Sebagai penerima sinyal bahwa DCE siap mengirimkan data
9	<i>Ring Indicator (RI)</i>	Sebagai pemberitahu telepon berbunyi

2.4 Perintah AT (AT Command)

Perintah AT (*Hayes AT Command*) digunakan untuk berkomunikasi dengan terminal (modem) melalui gerbang serial pada komputer. Dengan penggunaan perintah AT, dapat diketahui atau dibaca kondisi dari terminal, seperti mengetahui kondisi sinyal, kondisi baterai, mengirim pesan, membaca pesan, menambah item pada daftar telepon, dan sebagainya. Pada Tabel 2.5 diperlihatkan beberapa jenis perintah AT yang berhubungan dengan penanganan pesan-pesan SMS.

Tabel 2.5 beberapa jenis perintah AT.

AT Command	Fungsi
AT + CMGS	Mengirim pesan
AT + CMGR	Membaca pesan
AT + CMGD	Menghapus pesan
AT + CNMI	Indikasi pesan baru

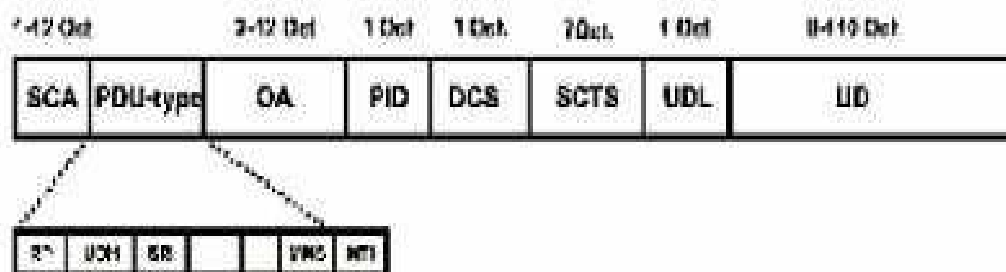
2.5 Protokol Data Unit (PDU)

Protocol Data Unit (PDU) merupakan salah 1 mode proses pengiriman atau penerimaan data (SMS) dari/ke *Message Center* (MC). Dalam mode PDU, pesan yang dikirim berupa informasi dalam bentuk data dengan beberapa kepala-kepala informasi. Hal ini akan memberikan kemudahan jika dalam pengiriman akan dilakukan kompresi data, atau akan dibentuk sistem penyandian data dari karakter dalam bentuk untaian bit-bit biner. PDU tidak hanya berisi pesan teks saja, tetapi terdapat beberapa meta-informasi yang lainnya, seperti nomor pengirim, nomor *SMS Centre*, waktu pengiriman, dan sebagainya. Semua informasi yang terdapat dalam PDU, dituliskan dalam bentuk pasangan-pasangan bilangan heksadesimal yang disebut dengan pasangan oktet. (Triyatmo Nugroho, 2008).

2.6 SMS Penerimaan (SMS-Deliver)

SMS Penerimaan (*SMS-Deliver*) adalah pesan yang diterima oleh terminal dari SMSC dalam bentuk PDU. PDU SMS-Penerimaan memiliki format seperti pada Gambar 2.8. Pada PDU ini, terdapat beberapa meta-informasi yang dibawa, antara lain:

- 1.7 *Service Centre Address* (SCA), Berisi informasi *SMS-centre*.
- 1.8 Tipe PDU (*PDU Type*), Berisi informasi jenis dari PDU tersebut.
- 1.9 *Originating Address* (OA) Berisi informasi nomor pengirim.
- 1.10 *Protocol Identifier* (PID) Berisi informasi Identifikasi Protokol yang digunakan.
- 1.11 *Data Coding Scheme* (DCS) Berisi informasi skema pengkodean data yang digunakan.
- 1.12 *Service Center Time Stamp* (SCTS) Berisi informasi waktu
- 1.13 *User Data Length* (UDL) Berisi informasi panjang dari data yang dibawa
- 1.14 *User Data* (UD) Berisi informasi data-data utama yang dibawa.

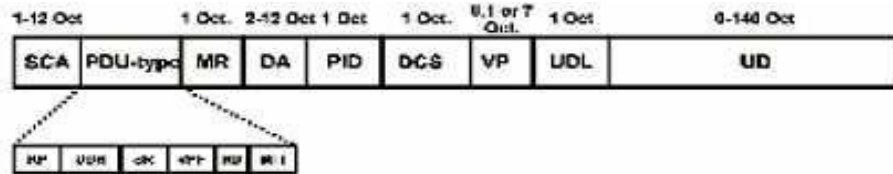


Gambar 2.8 Format PDU Penerima.

2.7 PDU Pengirim (PDU-Submit)

PDU Pengiriman memiliki informasi-informasi yang sama dengan PDU Penerimaan, sementara yang berbeda adalah berupa informasi (lihat Gambar 2.9):

- Message Reference* (MR), Parameter yang mengindikasikan nomor referensi SMS-Pengiriman
- Destination Address* (DA), Berisi informasi nomor alamat yang dituju.
- Validity Period* (VP), Berisi informasi jangka waktu validitas pesan pada jaringan.



Gambar 2.9 Format PDU Pengirim.

Pesan yang dikirim dalam mode PDU, berupa informasi dalam bentuk data dengan beberapa kepala-kepala informasi. Hal ini akan memberikan kemudahan jika dalam pengiriman akan dilakukan kompresi data, atau akan dibentuk sistem penyandian data dari karakter dalam bentuk susunan bit-bit biner. PDU tidak hanya berisi pesan teks saja, tetapi terdapat beberapa meta-informasi yang lainnya, seperti nomor pengirim, nomor *SMS Centre*, waktu pengiriman, dan sebagainya. Semua informasi yang terdapat dalam PDU, dituliskan dalam bentuk pasangan-pasangan bilangan heksadesimal yang disebut dengan pasangan *oktet*. *Octet* PDU berisi tidak hanya pesan, tetapi juga banyak informasi tentang pengirimnya seperti *Message Center* (SMSC), waktu dan lain-lain. Semuanya dalam heksadesimal *Octet* atau sistem desimal *semi-oktet*. *Octet-oktet* berikut adalah pesan SMS pengodean 7-bit dengan nomer SMSC +62855000000 dan pengirim +6285640614121 yang berisi kata 'hello' ;

07912658050000F011000D91265846604121F1000080703021223105E8329BFD06.

Tabel 2.6 Bagian-bagian format SMS

Oktet / Digit Hexa	Keterangan
01	Panjang atau jumlah pasangan digit dari nomor <i>service number</i> (SMSC) yang digunakan, dalam hal ini adalah 7 pasangan (14 digit berikutnya)
91	Jenis nomor SMSC. Angka 91 menandakan format nomor internasional (misal +6281xxx). Untuk 081xxx menggunakan angka 81.
2658050000F0	Nomor SMSC yang digunakan. Karena jumlah digit nomor SMS adalah ganjil, maka digit paling belakang dipasangkan dengan huruf F. Kalau diterjemahkan, nomor SMSC yang digunakan adalah +62855000000 (IM3)
11	Oktet pertama untuk pesan SMS yang diterima
0D	Panjang digit dari nomor pengirim (0D hex = 13 desimal)
91	Jenis nomor pengirim (sama dengan jenis nomor SMSC)

265846604121F1	Nomor pengirim SMS, yang jika diterjemahkan adalah +6285640614121 (format prefik internasional kalau di indonesia 0 diganti 62, F adalah nilai tambahan di akhir digit agar jumlah digitnya genap 14, kemudian setiap pasang karakte dibalik)
00	Pengenal protokol, dalam hal ini adalah 0
00	Skema pengkodean SMS, juga bernilai 0
807030 21223180	Waktu pengiriman, yang berarti 08-07-03 (3 Juli 2008), dan jam 12:22:13. Sedangkan 80 adalah <i>Timezone</i> yang digunakan.
05	Panjang dari pesan SMS, dalam hal ini adalah 4 huruf (dalam mode 7 bit).
E8329BFD06	Pesan SMS dalam mode 7 bit. Jika diterjemahkan kedalam 8 bit, lalu dirubah ke ASCII, maka didapat pesan 'hello'

Dalam pengkodean ini hanya mempunyai perbedaan di isi SMS saja. Pada Tabel 2.6 diatas merupakan pengkodean 7-bit. Pengkodean secara 8-bit isi setiap octet mewakili 1 karakter huruf/ angka/symbol. Misal, isi SMS "hello" dikodekan secara 8-bit menjadi "68656C6C6F" dan pengkodean 16-bit isi setiap dua octet mewakili 1 karakter huruf/angka/symbol. Misal, isi SMS "hello" dikodekan "68656C6C6F secara 16-bit menjadi 00680065006C006C006F". (Triyatmo Nugroho, 2008).

2.8 HP (*Handphone*) (*User*, *Sistem*)

2.8.1 HP (*Handphone*) *User*

HP *User* yaitu *handphone* yang berada pada sisi *user* untuk memantau sistem ini. *User* dapat memberikan inputan kepada sistem mikrokontroller yang ada di kendaraan dari jauh menggunakan SMS. *Short Message Service* (SMS) adalah fasilitas yang dimiliki oleh jaringan GSM (*Global System for Mobile communication*) yang memungkinkan pelanggan untuk mengirimkan dan menerima pesan-pesan singkat sepanjang 160 karakter. SMS ditangani oleh jaringan melalui suatu pusat layanan atau *SMS Service Center* (SMS SC) yang berfungsi untuk menyimpan dan meneruskan pesan dari sisi pengirim ke sisi penerima. Format SMS yang dipakai oleh produsen MS (*Mobile Station*) adalah *Protocol Description Unit* (PDU). Format PDU akan mengubah *septet-septet* kode ASCII (7 bit) menjadi bentuk *byte* PDU (8 bit) pada saat pengiriman data dan akan diubah kembali menjadi kode ASCII pada saat diterima oleh MS. PDU untuk mengirim SMS terdiri atas delapan *header*.

2.8.2 HP (*Handphone*) *Sistem*

HP sistem merupakan *handphone* yang terhubung pada rangkaian mikrokontroller untuk menerima dan mengeksekusi masukan dari HP *user*. Komunikasi antara mikrokontroller dengan HP sistem menggunakan kabel data dengan standard RS-232. Untuk mengubah *level* tegangan TTL mikrokontroller menjadi *level* tegangan RS-232 HP menggunakan IC MAX-232.

(Ilma Firmawan, 2008).

2.9 Relay

Relay merupakan rangkaian yang bersifat elektronis sederhana dan tersusun oleh :

- 2) Saklar
- 3) Medan elektromagnet (kawat koil)
- 4) Poros besi

Cara kerja komponen ini dimulai pada saat mengalirnya arus listrik melalui koil, lalu membuat medan magnet sekitarnya merubah posisi saklar sehingga menghasilkan arus listrik yang lebih besar. Di sinilah keutamaan komponen sederhana ini yaitu dengan bentuknya yang minimal bisa menghasilkan arus yang lebih besar.

Komponen sederhana ini dalam perkembangannya digunakan (*atau* pernah digunakan) sebagai komponen dasar berbagai perangkat elektronika, lampu kendaraan bermotor, jaringan elektronik, televisi, radio, bahkan pada tahun 1930-an pernah digunakan sebagai perangkat dasar komputer yang keberadaannya kini digantikan oleh mikroprosesor seperti IntelCorp. dan AMD. Keuntungan dalam pemakaian relay antara lain sebagai berikut:

- H. Dapat mengontrol arus secara otomatis.
- I. Dapat memaksimalkan besarnya tegangan listrik
- J. Pemakaian saklar / koin bisa lebih dari satu.



Gambar 2.10 Gambar fisik Relay
(www.wikipedia/relay.html, 2008)

2.10 Pengenalan Bahasa Pemrograman Assembler

Bahasa Assembler merupakan bahasa pemrograman tingkat rendah karena dalam proses pemrograman menggunakan bahasa mesin atau peralatan yang diprogram. Bahasa ini banyak digunakan dalam pemrograman mikrokontroller karena memiliki keunggulan dalam hal kecepatan, ukuran file yang kecil dan kemudahan dalam memanipulasi sistem komputer (S'to dalam Dyah E Indriyani, 2008).

Bahasa Assembler mempunyai beberapa instruksi yang digunakan dalam pemrograman, antara lain :

- a. **Instruksi aritmatika**, instruksi ini digunakan untuk melakukan operasi aritmatika yaitu penjumlahan, pengurangan, perkalian dan pembagian. Instruksi-instruksi aritmatika ditunjukkan oleh Tabel 2.7.

Tabel 2.7. Daftar Instruksi Aritmatika Mikrokontroller.

Mnemonic	Operasi	Mode-mode pengalaman
		Dir Ind Reg Imm
ADD A,<byte>	A=A+byte	x x x x

ADDC	A,<byte>	A=A+byte+C	x x x x
SUBB	A,<byte>	A=A-byte-C	x x x x
INC	A	A=A+1	Akkumulator saja
INC	<byte>	<byte>=byte+1	x x x
INC	DPTR	DPTR= DPTR+1	Data pointer saja
DEC	A	A=A-1	Akkumulator saja
DEC	<byte>	<byte>=byte-1	x x x
MUL	AB	B:A=BxA	ACC dan B saja
DIV	AB	A=int[A/B] B=mod[A/B]	ACC dan B saja
DA	A	Decimal Adjust	Akkumulator saja

- b. **Instruksi Logika**, instruksi logika ini digunakan untuk melakukan operasi *boolean* (AND, OR, XOR, NOT) pada suatu *byte* dan operasi pada masing-masing bit. Instruksi yang digunakan antara lain ditunjukkan pada Tabel 2.8. berikut :

Tabel 2.8. Daftar Instruksi Logika Mikrokontroler.

Mnemonic	Operasi	Mode-mode pengalamatan
		Dir Ind Reg Imm
ANL A,<byte>	A=A .and. <byte>	x x x x
ANL <byte>,#data	<byte>=<byte> .and. #data	x
ORL A,<byte>	A=A .or. <byte>	x x x x
ORL <byte>,#data	<byte>=<byte> .or. #data	x
XRL A,<byte>	A=A .xor. <byte>	x x x x
XRL <byte>,#data	<byte>=<byte> .xor. #data	x
CLR A	A = 00h	Akumulator saja
CPL A	A = .not. A	Akumulator saja
RL A	Putar ACC kiri 1 bit	Akumulator saja
RLC A	Putar kiri lewat carry	Akumulator saja
RR A	Putar ACC kanan 1 bit	Akumulator saja
RRC A	Putar kanan lewat carry	Akumulator saja
SWAP A	Tukar 4-bit dalam A	Akumulator saja

- c. **Instruksi Transfer Data**, instruksi ini digunakan untuk melakukan transfer data yang mengakses ruang memori data *internal* serta mode pengalamatan yang terkait untuk pemindahan data antar lokasi di dalam ruang memori *internal*. Instruksi yang digunakan antara lain ditunjukkan oleh Tabel 2.9. berikut:

Tabel 2.9. Daftar Instruksi Transfer Data Mikrokontroler.

Mnemonic	Operasi	Mode-mode pengalamatan
		Dir Ind Reg Imm
MOV A,<number>	A = <sumber>	x x x x
MOV DPTR,#data16	DPTR=konstanta 16-bit	x x

MOVX A,@DPTR	Membaca RAM external lokasi DPTR	
MOVC A,@A+DPTR	Membaca memori program dilokasi	
PUSH <sumber>	INC SP MOV ”@Sp”, <sumber>	x
POP <tujuan>	MOV <tujuan>, ”@SP” DEC SP	x
XCH A,<byte>	Tukar data <i>byte</i> antara ACC dan <byte>	x x x x
XCHD A,@Ri	Tukar data nibel rendah antara ACC dan @Ri	

- d. **Instruksi Boolean**, instruksi ini digunakan untuk memeriksa suatu lokasi bit apakah berisi 1 atau 0. Beberapa contoh instruksi ditunjukkan oleh Tabel 2.10 berikut:

Tabel 2.10. Daftar Instruksi Boolean Mikrokontroller.

Instruksi	Fungsi
ANL C,bit	C = C .and. bit
ANL C,/bit	C = C .and. .not. bit
ORL C,bit	C = C .or. bit
ORL C,/bit	C = C .or. .not. bit
MOV C,bit	C = bit
CLR C	C = 0
CLR bit	bit = 0
SETB C	C = 1
SETB bit	bit = 1
CPL C	C = .not. C
CPL bit	bit = .not. bit
JC rel	Jump if C = 1
JNC rel	Jump if C = 0
JB bit,rel	Jump if bit = 1
JNB bit,rel	Jump if bit = 0
JBC bit,rel	Jump if bit = 1, CLR bit

- e. **Instruksi Lompat**, instruksi ini digunakan untuk melakukan lompat ke variabel lain jika syarat instruksi telah dipenuhi. Instruksi yang digunakan antara lain seperti yang ditunjukkan oleh Tabel 2.11. berikut :

Tabel 2.11. Daftar Instruksi Lompat Mikrokontroller.

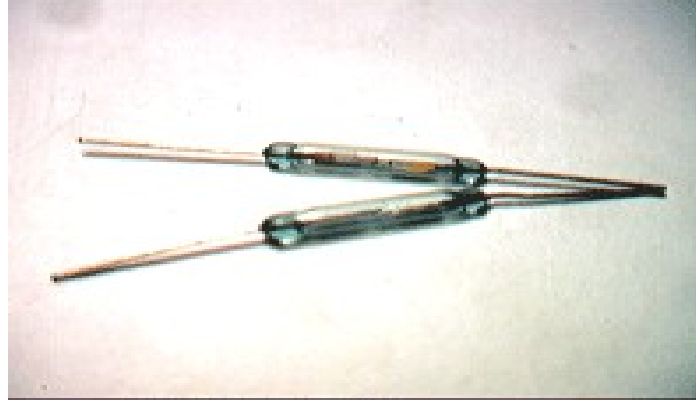
Instruksi	Fungsi
JMP alamat	Lompat ke alamat
JMP @A+DPTR	Lompat ke alamat DPTR
CALL alamat	Panggil subrutin di alamat
RET	Kembali dari subrutin
RETI	Kembali dari layanan interupsi

(Dyah E Indriyani, 2008).

2.11 Sensor *Reed Switch*

Reed sensor digunakan untuk mendeteksi adanya medan magnet, *reed switch* adalah suatu komponen elektronika yang sangat sederhana dengan menggunakan dua buah kawat untuk saling menghubungkan. keuntungan dengan menggunakan *reed switch* adalah ujung kontaktor telah

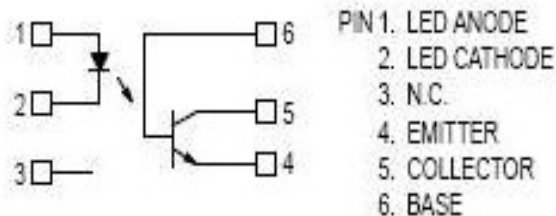
terindungi dengan aman dari lingkungan luar. *Reed switch* digunakan untuk mendeteksi medan magnet, *reed switch* mempunyai *default normaly open*, tetapi saat didekatkan pada magnet, maka akan *close*. *Reed switch* biasanya menggunakan magnet untuk membuka atau menutup circuit. Bentuk *reed switch* adalah *potential divider* dimana tegangan output yang ditentukan oleh *upper* dan *lower* pada rangkaian. Rangkaian untuk sensor reed switch tergantung pada sensor yang dihubungkan ke sensor unit atau *inverted sensor*, saat terhubung ke sensor unit.



Gambar 2.11. *Reed switch*.
(<http://azies-tech.com/>, 2009)

2.12 Optocoupler

Optocoupler merupakan salah satu jenis komponen yang memanfaatkan sinar sebagai pemicu *on/off*-nya. *Opto* berarti optik dan *coupler* berarti pemicu. Sehingga bisa diartikan bahwa optocoupler merupakan suatu komponen yang bekerja berdasarkan picu cahaya optic opto-coupler termasuk dalam sensor, dimana terdiri dari dua bagian yaitu cahaya (*transmitter*) dan deteksi sumber cahaya (*receiver*). Penggunaannya memungkinkan untuk memisahkan dua bagian dengan tegangan kerja berbeda.



Gambar 2.12. Pin Optocoupler M3020.

Bagian pemancar atau *transmitter* dibangun dari sebuah led infra merah untuk mendapatkan ketahanan yang lebih baik daripada menggunakan led biasa. Sensor ini bisa digunakan sebagai isolator dari rangkaian tegangan rendah ke rangkaian tegangan tinggi. Selain itu juga bisa dipakai sebagai pendeteksi adanya penghalang antara *transmitter* dan *receiver* dengan memberi ruang uji dibagian tengah antara led dengan *photo transistor*. Ada berbagai macam bentuk, jenis, dan tipe optocoupler seperti MOC 3040 atau 3020, 4N25 atau 4N33 dan sebagainya.

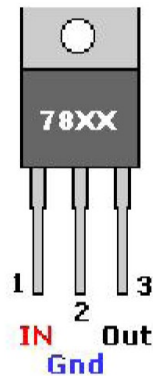
Pada umumnya semua jenis optocoupler pada lembar datanya mampu dibebani tegangan sampai 7500 Volt tanpa terjadi kerusakan atau kebocoran. Di pasaran optocoupler tersedia dengan type 4NXX atau MOC XXXX dengan X adalah angka part valuenya. Untuk type 4N25 ini mempunyai tegangan isolasi sebesar 2500 Volt dengan kemampuan maksimal led dialiri arus *forward* sebesar 80 mA. Namun besarnya arus led yang digunakan berkisar antara 15mA - 30 mA dan untuk menghubungkannya dengan tegangan +5 Volt diperlukan tahanan pembatas.

([http:// elektronika-elektronika.blogspot.com](http://elektronika-elektronika.blogspot.com), 2009).

2.13 IC LM7805

IC LM7805 adalah IC regulator untuk mendapatkan tegangan keluaran sebesar 5 volt. Selain LM7805, IC regulator lain yang terdapat di pasaran antara lain IC7812 yang berfungsi sebagai regulator tegangan 12 volt, sedangkan seri 79XX misalnya adalah 7905 dan 7912 yang berturut-turut adalah regulator dengan tegangan keluaran adalah negatif 5 volt dan 12 volt.

Selain dari regulator tegangan tetap ada juga IC regulator yang tegangannya dapat diatur. Prinsipnya sama dengan regulator OP-amp yang dikemas dalam satu IC misalnya LM317 untuk regulator variable positif dan LM337 untuk regulator variabel negatif. Beda Resistor R1 dan R2 ada di luar IC, sehingga tegangan keluaran dapat diatur melalui resistor *eksternal* tersebut. Bentuk fisik dari IC LM7805 dapat dilihat pada Gambar 2.13 berikut.



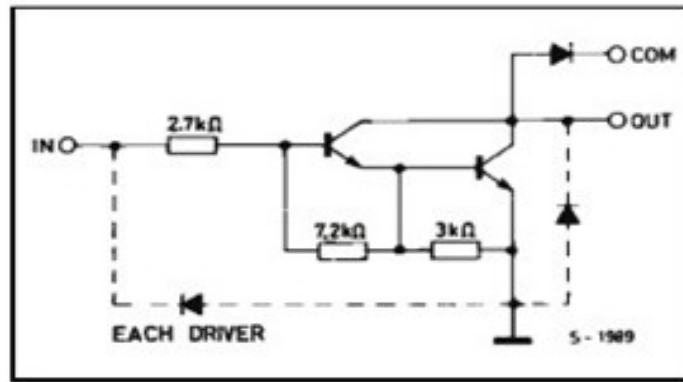
Gambar 2. 13 IC LM7805

2.14 IC ULN 2803

IC ULN2803 adalah transistor *array* yang berisi 8 buah transistor NPN yang masing masing berkonfigurasi Darlington. Seluruh emitor dari semua darlington dihubungkan menjadi satu pada pin GND. Masing-masing darlington dilengkapi dengan *freewheeling* diode/integral-suppression diode. *Freewheeling* diode akan melindungi darlington dari ggl balik (*back emf*) saat dihubungkan dengan beban induktif (dapat berupa relay, solenoida, motor listrik atau transformator). Katoda dari *freewheeling* diode tersebut dihubungkan menjadi satu dan biasanya dihubungkan dengan catu positif catu daya. ULN 2803 didesain untuk dapat digunakan bersama-sama dengan rangkaian IC TTL. Tiap darlington pada ULN2803 memiliki rating arus 500 mA dan rating tegangan 50V pada saat *cut-off*. Tiap darlington pada ULN2803 memiliki hFE minimal 1000 hFE.

ULN2803 bukanlah IC TTL sebagaimana IC TTL 74LS07. ULN2803 dikendalikan dengan memberikan arus pada masukannya, sedangkan 74LS07 dikendalikan dengan memberikan tegangan TTL (0–5V) pada masukannya. Perhitungan arus basis dan arus kolektor perlu dilakukan sebagaimana perhitungan sinyal besar pada rangkaian transistor *common emitter*. ULN2803 memiliki RB internal sebesar 2.7kΩ.

ULN2803 tidak memiliki RC internal. Saat saturasi, tiap darlington pada ULN 2803 memiliki tegangan VCE maksimal 1.3V. Seperti penggunaan transistor *open collector* biasa, ULN 2803 tidak memerlukan catu daya tersendiri untuk mengoperasikannya. Tiap darlington pada ULN 2803 dioperasikan dengan memberikan arus basis minimal 0.93 mA. Konfigurasi tiap darlington pada ULN2803 dapat dilihat pada Gambar 2.14 di bawah ini.

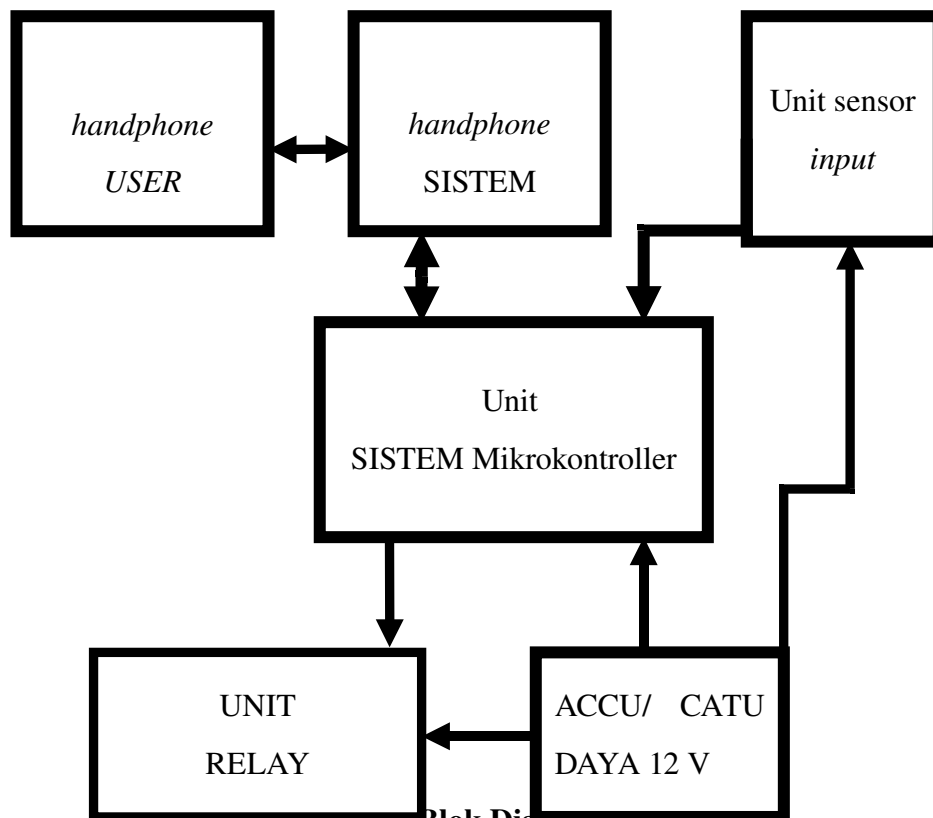


Gambar 2.14. Konfigurasi tiap darlington pada ULN2803.
(Donny Eryanto, 2009).

BAB III DESAIN DAN PERANCANGAN

a) Blok Diagram Sistem

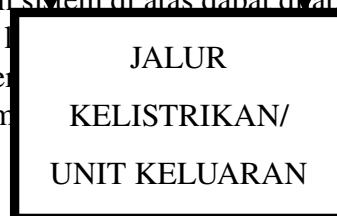
Sebelum pembuatan alat / perangkat keras langkah pertama yang dilakukan adalah pembuatan dan pemahaman tentang blok diagram sistem. Secara keseluruhan blok diagram sistem dapat ditunjukkan oleh Gambar 3.1 berikut.

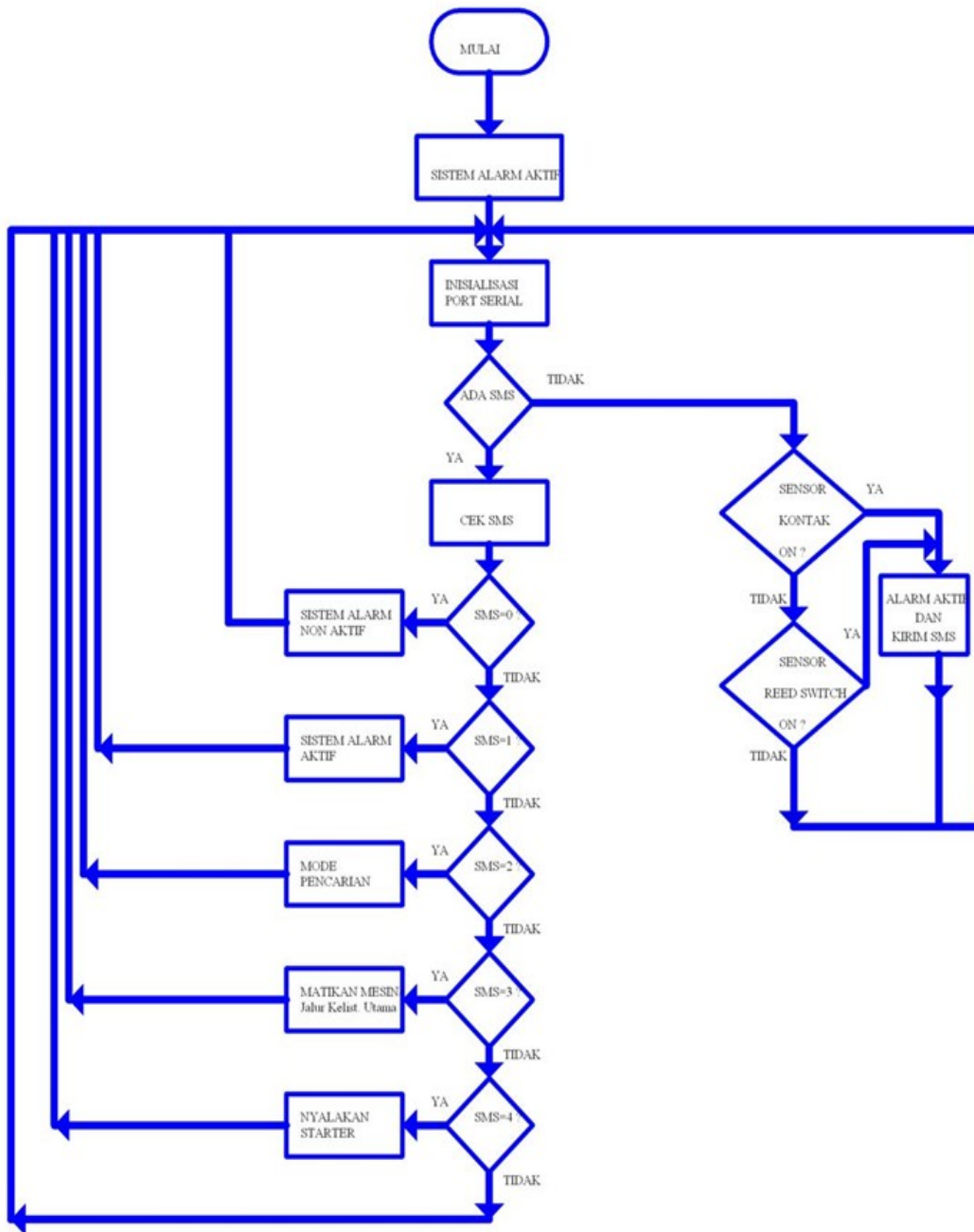


Gambar 3.1 Blok Diagram Sistem

b) Diagram Alir Sistem

Dari blok diagram sistem di atas dapat dijabarkan dengan gambar diagram alir sistem yang menunjukkan prinsip kerja dari sistem *warning* sepeda motor berikut ini ditunjukkan pada Gambar 3.2





Gambar 3.2 Diagram Alir Sistem

c) Proses Kerja Sistem

Pada tugas akhir ini akan dilakukan rancang bangun suatu sistem *warning* pada sepeda motor jarak jauh berbasis SMS dengan mikrokontroller AT89S51 sebagai pengendali utama dari sistem tersebut.

Sistem ini akan bekerja begitu alat dihubungkan dengan AKI Sepeda motor. Kemudian sistem akan melakukan inisialisasi *interrupt port* serial yang terhubung dengan *handphone* sistem untuk mendeteksi apabila sistem mikrokontroller mendapat *input-an* (perintah SMS) yang berasal dari *handphone user* dan sistem akan melakukan inisialisasi terhadap *interrupt* yang berasal dari sensor kontak dan sensor kemudi (*reed switch*). Apabila *interrupt port* serial aktif berarti sistem menerima perintah SMS yang berasal dari *handphone* sistem. Perintah SMS yang diproses oleh *handphone* sistem dibatasi hanya untuk perintah SMS yang berasal dari *handphone user*, dimana

handphone user memiliki nomer SIM card yang telah teridentifikasi oleh program. Perintah SMS yang diproses dapat dilihat pada Tabel 3.1 berikut.

Tabel 3.1 Instruksi SMS

PERINTAH SMS	FUNGSI
0	MENONAKTIFKAN SISTEM ALARM
1	MENGAKTIFKAN SISTEM ALARM
2	MENGAKTIFKAN MODE PENCARIAN
3	MEMATIKAN MESIN KENDARAAN
4	MENYALAKAN STARTER

Selain perintah pada Tabel 3.1 di atas maka, setiap perintah SMS yang masuk ke *handphone* sistem langsung dihapus, sehingga tidak diproses untuk menjalankan sistem.

Selain melakukan inisialisasi terhadap *interrupt* serial, sistem juga melakukan pendeteksian terhadap *interrupt* yang berasal dari sensor kontak dan sensor kemudi (*reed switch*). Apabila salah satu *interrupt* yang berasal dari sensor kemudi atau sensor kontak aktif, serta sistem alarm juga dalam keadaan aktif, maka sistem akan membunyikan alarm / klakson kendaraan dan sistem juga akan mengirimkan SMS peringatan kepada *handphone user*. Tabel 3.2 berikut merupakan Tabel SMS peringatan dari *handphone* sistem ke *handphone user* apabila terdapat pengaktifan dari sensor.

Tabel 3.2 SMS Peringatan

SENSOR AKTIF	SMS PERINGATAN
KEMUDI (<i>REED SWITCH</i>)	Sensor Kemudi ON
KONTAK	Sensor Kontak ON

Sistem *warning* sepeda motor ini juga dilengkapi dengan fasilitas tambahan yaitu dapat menghidupkan starter kendaraan dari jarak jauh melalui perintah SMS dari *handphone user* dengan mengirimkan perintah SMS ('4') dari *handphone user* ke *handphone* sistem. Sistem juga dilengkapi dengan menu atau *feature* pencarian untuk memudahkan pemilik (*user*) menemukan kendaraannya saat berada di parkiran. Menu ini dapat diaktifkan dengan cara mengirimkan perintah SMS ('2') ke *handphone* sistem.

d) Analisis Kebutuhan

Pembuatan alat pengaman sepeda motor jarak jauh dengan memanfaatkan fasilitas dan SMS ini memerlukan beberapa sarana yang digunakan untuk membangun sistem tersebut. Beberapa hal yang dibutuhkan antara lain:

xxiv. Perangkat Keras (*hardware*)

Beberapa perangkat keras yang digunakan untuk pembuatan alat pengaman sepeda motor ini antara lain :

A. ACCU Sepeda Motor atau Catu Daya 12 Volt DC

Catu daya yang digunakan untuk menyuplay daya ke sistem pengaman ini adalah dari AKI (accu) sepeda motor itu sendiri. AKI (accu) sepeda motor biasanya mempunyai tegangan sumber 12 Volt. Tegangan ini dibutuhkan untuk menjalankan relay yang terhubung dengan sistem mikrokontroller untuk memutus atau menyambung jalur kelistrikan pada sepeda motor, tegangan ini jugalah yang disaring melalui IC regulator LM7805 menjadi 5 Volt DC. Tegangan 5 Volt inilah yang nantinya dialirkan ke rangkaian sistem

B. Unit Sensor *Input*

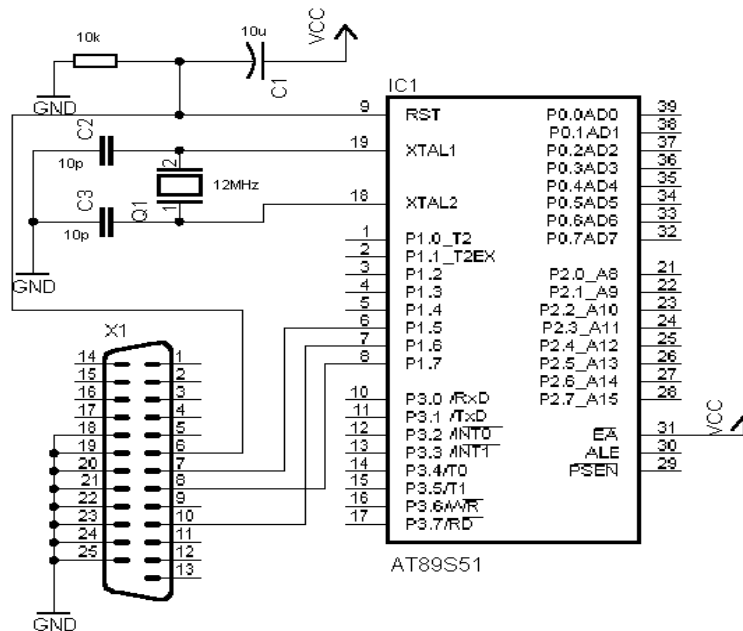
- i) Sebuah sensor *reed switch* yang terpasang pada bagian leher kendaraan atau pada bagian pengunci stang. Sistem kerja dari sensor ini adalah apabila sensor didekatkan dengan magnet maka switch akan bekerja sehingga akan mengalirkan arus menuju optocoupler yang terhubung dengan rangkaian mikrokontroler, dan kemudian akan dijadikan sinyal *input*-an untuk menjalankan sistem alarm yang telah diprogram.
- ii) Sebuah sensor kontak. Sensor ini hanya merupakan suatu rangkaian yang dihubungkan dengan kontak kendaraan, dimana apabila kontak kendaraannya aktif maka akan memberikan masukan pada sistem mikrokontroler untuk nantinya diolah dan dijadikan keluaran untuk menjalankan sistem alarm yang telah diprogram.
- iii) Sebuah sensor gigi netral. Sensor ini merupakan rangkaian yang dihubungkan dengan indikator gigi netral yang terdapat pada kendaraan. Apabila indikator gigi netral nyala atau aktif, maka akan dijadikan sinyal masukan untuk rangkaian mikrokontroler, kemudian sinyal tersebut akan diproses untuk menghidupkan starter kendaraan melalui jarak jauh (SMS). Apabila indikator sensor gigi netral tersebut tidak aktif maka sepeda motor tidak dapat dihidupkan melalui jarak jauh (melalui pengiriman perintah SMS).

Unit sistem mikrokontroller tersusun dari beberapa komponen elektronik dan juga chip mikrokontroller, komponen-komponen tersebut antara lain :

1. Kapasitor
 - e. 30 pF 2 buah
 - f. 2,2 μ F 2 buah
 - g. 4,7 μ F 1 buah
 - h. 10 μ F 5 buah
 - i. 220 μ F 1 buah
2. Resistor
 - j. R (22 K) 1 buah
 - k. RP (10 K) 2 buah
3. IC (*Integrated Circuit*)
 - l. 7805 1 buah
 - m. MAX232 1 buah

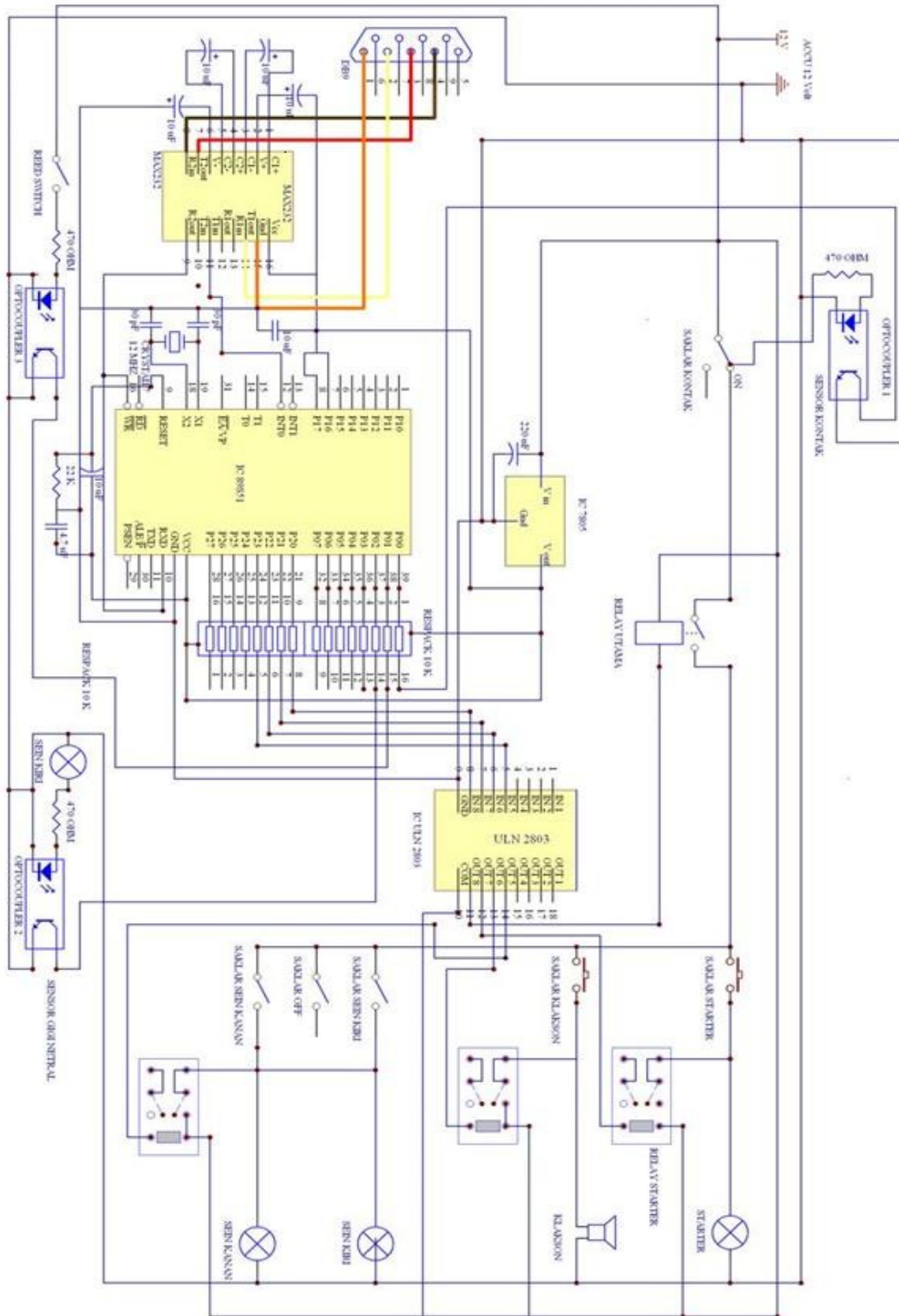
- | | |
|---------------------------------|--------|
| n. ULN2803 | 1 buah |
| 4. Chip Mikrokontroller AT89S51 | 1 buah |
| 5. Crystal/ XTI 110592 | 1 buah |
| 6. Konektor | |
| o. IC 40 pin | 1 buah |
| p. IC 18 pin | 1 buah |
| q. IC 16 pin | 1 buah |

Perlu diketahui pula bahwa mikrokontroller tersebut bekerja dengan memasukan program yang berbentuk file heksa ke dalamnya. Program yang digunakan dalam pembuatan tugas akhir ini adalah ASM 51 dan AEC ISP. Bahasa yang digunakan adalah bahasa *assembler*. Bahasa *assembler* yang telah jadi dan yang berisi instruksi-instruksi diubah ke dalam file heksa dengan menggunakan ASM 51 dan kemudian memasukan file heksa ke mikrokontroller. Proses memasukkan file heksa tersebut menggunakan perangkat lunak yang dinamakan AEC ISP. Perangkat lunak tersebut merupakan kesatuan dari apa yang dinamakan dengan *downloader*. Rangkaian dari *downloader* tersebut seperti terdapat pada Gambar 3.2 dibawah ini.



Gambar 3.2. Rangkaian Downloader

Rangkaian Sistem mikrokontroller dapat dilihat pada Gambar 3.4. berikut ini, atau pada lampiran 1 halaman 68.



Gambar 3.4. Skema rangkaian sistem *warning* sepeda motor berbasis SMS

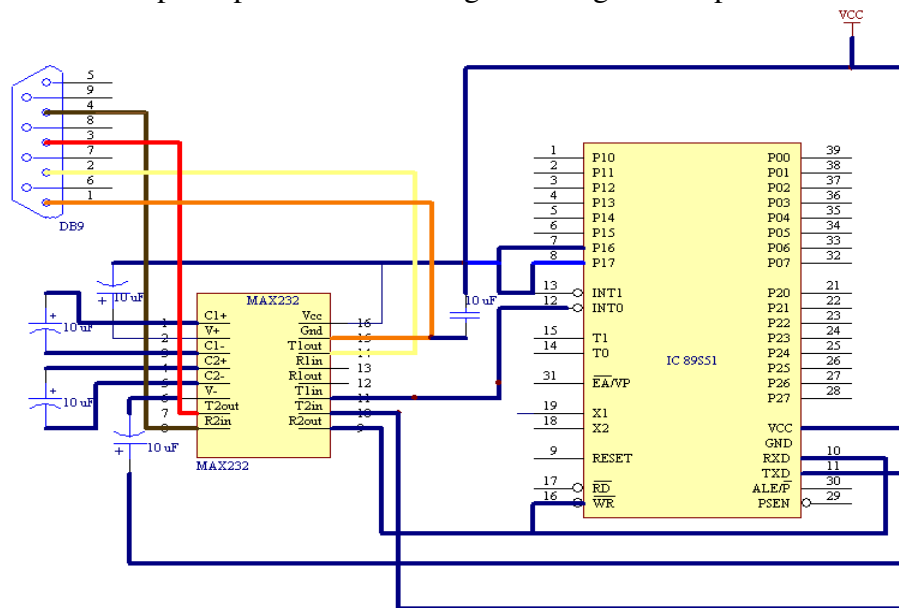
D. Handphone

Handphone disini adalah suatu alat yang difungsikan sebagai pemancar dan sekaligus sebagai penerima informasi dari sistem mikrokontroller maupun dari perintah

SMS yang dikirimkan dari *handphone* pemilik kendaraan. Fasilitas SMS dapat ditemukan pada *handphone* komersil. Pada sistem *warning* sepeda motor jarak jauh dengan teknologi *Short Message Service* (SMS) ini memanfaatkan *handphone* Siemens seri 25/35/45 sebagai alat pengirim maupun penerima SMS. *Handphone* tersebut mempunyai I/O data serial yang dapat digunakan untuk kendali pengiriman SMS melalui perintah AT (AT command).

E. Antarmuka Serial

Komunikasi data serial yang digunakan untuk komunikasi antara HP dengan Mikrokontroler adalah standard RS-232, maka perlu dilakukan pengubahan tegangan ke level TTL. Proses pengubahan dilakukan dengan menggunakan IC MAX 232 dengan tambahan beberapa kapasitor 1 uF dengan konfigurasi seperti Gambar 3.5. dibawah ini.

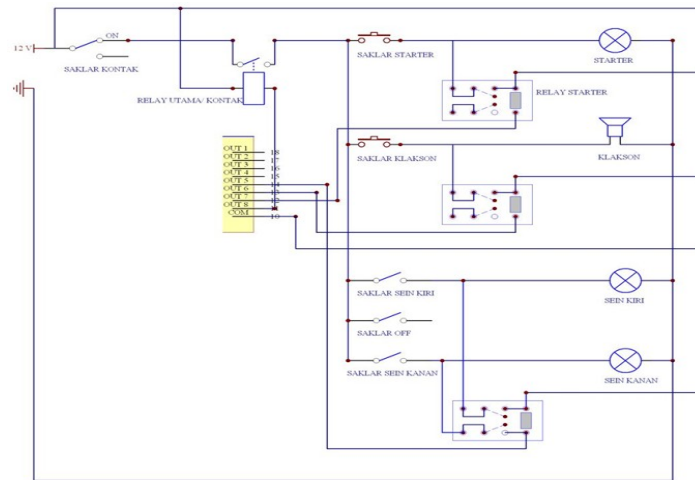


Gambar 3.5. Rangkaian untuk Komunikasi Serial Menggunakan IC MAX232

F. Unit Keluaran dan relay

Unit keluaran pada sistem ini merupakan suatu rangkaian yang telah terpasang pada sepeda motor. Unit keluaran yang akan diatur kinerjanya dalam sistem ini diantaranya adalah klakson, lampu dan sein, starter elektrik, dan pemicu *ON/ OFF* pada kontak kendaraan. Masing-masing unit keluaran tersebut dihubungkan dengan unit relay yang terhubung dengan sistem mikrokontroler, sehingga nantinya unit keluaran dapat diatur melalui sistem mikrokontroler yang telah diprogram.

Unit relay adalah suatu rangkaian yang digunakan untuk mengatur kinerja dari unit keluaran. Unit ini terdiri dari 4 buah relay, dimana keempat relay tersebut dihubungkan dengan sistem mikrokontroler dan komponen-komponen keluaran. Salah satu dari relay yang digunakan dalam sistem ini harus memiliki nilai kuat arus tinggi, karena nantinya relay tersebut akan dilalui oleh arus yang digunakan untuk mengaktifkan semua komponen kelistrikan pada sepeda motor yaitu dihubungkan dengan rangkaian pemicu *ON/OFF* pada kontak kendaraan. Gambar 3.6 Berikut ini adalah skema dari Unit relay yang akan dihubungkan dengan masing-masing unit keluaran dan juga dengan dengan *port-port* yang ada pada mikrokontroler. Gambar skema rangkaian ini juga dapat dilihat pada lampiran 2 halaman 69.



Gambar 3.6. Skema rangkaian keluaran dan relay

xxv. Perangkat Lunak (*Software*).

Berikut adalah *Source Code* program yang digunakan dalam sistem *warning* sepeda motor jarak jauh berbasis SMS. Listing program (*Source Code*) dapat juga dilihat pada Lampiran 3 halaman 70.

;sistem *warning* SMS

;=====

\$mod51

skontak bit p0.0 ; aktif low

sreed bit p0.1

snetral bit p0.2

smesin bit p0.3

relay equ p2

rkontak bit p2.0 ; aktif high

rstart bit p2.1

rbel bit p2.2

rsign bit p2.3

dtr bit p3.2

falarm bit p3.5

connect bit p3.7

dseg at 20h

flag: ds 1

o_pdu: ds 1

dthp: ds 90

cseg at 0h

jmp start

org 0bh ; timer 0 interrupt

jmp timer0

org 23h ; serial interrupt

```

        jmp    serin

start:   mov    relay,#0
        mov    tmod,#21h    ; t1 8 bit autoreload,
        mov    th1,#0fdh    ; 9600 BPS
        orl    pcon,#10000000b ; SMOD=1 --> 19200 BPS
        setb   tr1          ; timer 1 run
        mov    scon,#50h    ; serial mode 1
        mov    ip,#02      ; priority timer 0 high
        setb   ea          ; enable all interrupt
        setb   es          ; enable serial
        setb   et0         ; enable timer 0
        clr    dtr         ; ignition on
        call   clr_ram
er_con:  call   delay
        mov    dptr,#p_ate0 ; kirim perintah
        call   command     ; AT
        call   delay
        mov    th0,#0
        setb   tr0
        jnb    f0,$         ; tunggu jawaban
        clr    f0
        mov    dptr,#j_ate0 ; bandingkan
        call   ceking      ; jawaban
        mov    a,flag
        cjne   a,#1,er_con  ; salah,--> ulang
        clr    connect     ; benar,--> led connect on
        mov    dptr,#p_cnmi
        call   command
        call   delay
        jnb    f0,$         ; tunggu jwb cnmi
        clr    tr0
        clr    f0
        jmp    csms
        ;*****
lagi:    jb     skontak,creed ; sensor kontak
        jnb    falarm,alarmof
        clr    rkontak      ; kontak off
        setb   rbel         ; bel on
        setb   rsign        ; sign on
        mov    dptr,#s0000
        call   sendsms
        call   dalarm
        clr    rbel
        clr    rsign
        jmp    creed

```

```

alarmof:  setb    rkontak    ; kontak on
creed:    jnb     sreel,csms  ; sensor      reed sw
          jnb     falarm,csms ; sistem off, cek sms
          setb    rbel       ; bel on
          setb    rsign      ; sign on
          mov     dptr,#s0001 ; kirim sms alarm
          call    sendsms
          call    dalarm
          clr     rbel
          clr     rsign
csms:     jnb     f0,lagi     ; tunggu sms

```

```

          clr     tr0
          clr     f0
          call    delay
          mov     dptr,#p_read ; kirim perintah
          call    command      ; baca sms
          call    delay
          jnb     f0,$          ; tunggu jawaban
          clr     tr0
          clr     f0
          call    proses       ; proses data
          call    clr_ram
          mov     dptr,#p_del1 ; kirim perintah
          call    command      ; hapus sms
          call    delay
          jnb     f0,$          ; tunggu jawaban
          clr     tr0
          clr     f0

          mov     dptr,#p_del2 ; kirim perintah
          call    command      ; hapus sms
          call    delay
          jnb     f0,$          ; tunggu jawaban
          clr     tr0
          clr     f0

          mov     dptr,#p_del3 ; kirim perintah
          call    command      ; hapus sms
          call    delay
          jnb     f0,$          ; tunggu jawaban
          clr     tr0
          clr     f0
          mov     dptr,#p_del4 ; kirim perintah
          call    command      ; hapus sms

```

```

        call    delay
        jnb     f0,$          ; tunggu jawaban
        clr     tr0
        clr     f0
        JMP     csms          ; lagi
;=====
;Clear Ram
;-----
clr_ram:  mov     r1,#dthp
          mov     b,#90
clear:    mov     @r1,#255    ; clear dthp
          inc     r1
          djnz    b,clear
          mov     dthp,#254
          ret
;=====
;Kirim perintah
;-----
command:  clr     f0
          mov     r1,#dthp
cmd:      mov     a,#0
          movc    a,@a+dptr
          cjne    a,#255,lj    ; 255 akhir data
          ret      ; selesai
lj:       clr     ti           ;
          mov     sbuf,a       ; kirim perintah
          jnb     ti,$
          inc     dptr         ; perintah selanjutnya
          jmp     cmd
;=====
;Interupsi Serial disimpan pada alamat r1
;-----
serin:    jnb     ri,xsr       ; rx?
          setb    tr0          ; ya--> matikan timer
          mov     r1,#dthp
slg:      mov     th0,#-30
          mov     @r1,sbuf     ; simpan
          clr     ri           ; rx ready
          inc     r1           ; alamat berikutnya
lsr:      jb      ri,slg
          jb      tr0,lsr
xsr:      reti
;=====
;Serial time out
;-----
timer0:   clr     tr0          ; timer off

```

```

MOV TH0,#-30
setb  f0
mov   @r1,#255
XTIM0:  reti
;=====
;Cek Jawaban HP; dptr=jml,referensi ; sama --> flag=1
;-----
ceking:    mov    flag,#1      ; flag benar
           mov    r1,#dthp
u_cek:     mov    a,#0
           movc   a,@a+dptra   ; jml referensi
           cjne   a,#255,l_cek
           ret
l_cek:     mov    b,@r1        ; ambil jwb hp
           cjne   a,b,ljtx     ;bandingkan
           inc    dptra        ; sama lanjutkan
           inc    r1
           jmp    u_cek        ; semua sama, --> flag=benar
ljtx:      mov    flag,#0      ; tidak sama, flag=salah -->selesai
           ret
;=====
;PDU; r1=data hp
;-----
proses:    mov    r1,#dthp+38  ; 36 ofset nomor pengirim
           mov    dptra,#no_hp ; bandingkan
pdu2:      mov    b,@r1
           ;call   cektx
           mov    a,#0
           movc   a,@a+dptra
           cjne   a,#255,pdu1
           jmp    pdu3        ; sama? --> lanjut
pdu1:      ;mov    b,a         ; ignore no hp
           cjne   a,b,xpdu     ; tidak? --> hapus
           inc    r1
           inc    dptra
           jmp    pdu2
pdu3:      mov    r1,#dthp+72  ; 72   ofset kode on/off
           ;MOV    A,@R1
           ;CALL   CEKTX
           ;INC    R1
           ;MOV    A,@R1
           ;CALL   CEKTX
           ;DEC    R1
           call    hex2
           anl    a,#01111111b ; ambil 7 bit
           ;call    cektx

```



```

        cjne    a,#'0',on1
        clr     falarm           ; alarmoff
        ;setb   rkontak
        ret
on1:      cjne    a,#'1',on2
        setb    falarm           ; alaram on
        clr     rkontak
        ret
on2:      cjne    a,#'2',on3      ; find on
        setb    rsign            ; sign on
        setb    rbel             ; bel on
        call    delay
        clr     rbel             ; bel off
        call    dalarm
        clr     rsign            ; sign off
        ret
on3:      cjne    a,#'3',on4
        clr     rkontak          ; mesin off
        ret
on4:      cjne    a,#'4',xpdu
        jnb     snetral,nostart
        setb    rkontak
        call    dstart
        setb    rstart           ; starter
        call    dstart
        clr     rstart
xpdu:     ret
nostart:  setb    rbel
        call    delay
        clr     rbel
        ret

;=====
;Kirim sms
;-----
sendsms:  push    dph
        push    dpl
        mov     b,#10            ; tunda 1 detik
ccst:     call    delay
        djnz    b,ccst
        mov     dptr,#p_send     ; kirim AT+cmgr=28
n_send:   mov     a,#0
        movc    a,@a+dptr
        cjne    a,#255,lsend
        jmp     s_pdu
lsend:    call    cektx
        inc     dptr

```

```

s_pdu:    jmp    n_send
          call   delay
          jnb    f0,$           ; tunggu jawaban
          clr    f0
          clr    tr0
          mov    b,#10         ; tunda 1 detik
ls_pdu:    call   delay
          djnz   b,ls_pdu
          pop    dpl
          pop    dph
ntxpdu:    mov    a,#0
          movc   a,@a+dptra
          cjne   a,#255,txpdu
          call   delay
          jnb    f0,$           ; tunggu jawaban
          clr    f0
          clr    tr0
          ret
txpdu:     call   cektx         ; kirim pesan
          inc    dptra
          jmp    ntxpdu

```

;=====

;conversi ascii ke hexadecimal

;-----

```

hex2:      call   asc2hex       ; digit 2
          mov    b,#16
          mul    ab
          push   acc
          inc    r1
          call   asc2hex
          mov    b,a
          pop    acc
          add    a,b
          inc    r1
          ret
asc2hex:    mov    a,@r1        ; digit 1
          clr    c
          subb   a,#30h
          cjne   a,#10,asc1
asc1:       jc    asc2
          clr    c
          subb   a,#7
asc2:       ret

```

;=====

;Tx Acc

;-----

```

cectx:      clr      ti
            mov      sbuf,a
            jnb      ti,$
            CLR      ti
            ret

;=====
;Delay
;-----
delay:      mov      r7,#0
dly:        mov      r6,#0
            djnz     r6,$           ; 256*2us=512uS
            djnz     r7,dly        ; 256*512uS=131.072uS
            ret              ; 0,13 S

Dalarm:     mov      r5,#200
dalarm1:    call     delay          ;0,13S
            djnz     r5,dalarm1    ;200*0,13=26S
            ret

Dstart:     mov      r5,#15
dstart1:    call     delay          ;0,13*15=1,95S
            djnz     r5,dstart1
            ret

;=====
;Tabel
;-----
p_ate0:     db 'ATE0',13,10,255
p_cnmi:     db 'AT+CNMI=1,1,2,2,1',13,10,255
p_read:     db 'AT+CMGR=1',13,10,255
p_del1:     db 'AT+CMGD=1',13,10,255
p_del2:     db 'AT+CMGD=2',13,10,255
p_del3:     db 'AT+CMGD=3',13,10,255
p_del4:     db 'AT+CMGD=4',13,10,255
no_hp:      db '265846178912F9',255
j_ate0:     DB 13,10,'OK',255

end

```

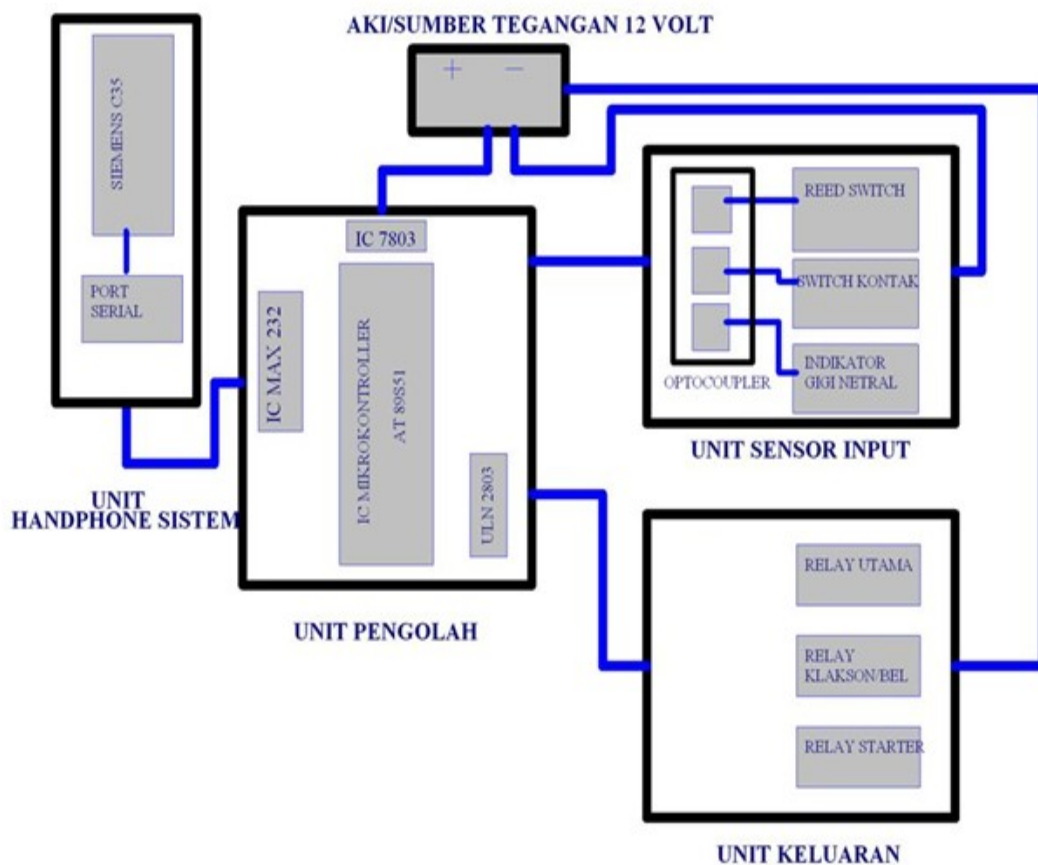
Setelah program selesai dibuat langkah selanjutnya adalah memasukan program tersebut kedalam IC mikrokontroler. Program dimasukan menggunakan *software* kompiler *AEC_ISP.EXE*. *File* yang bisa dimasukan kedalam mikrokontroler adalah *file* dengan ekstensi *.HEX* hasil dari *ASM51.EXE*. Apabila program telah sukses dan tidak terdapat error berarti mikrokontroller telah siap digunakan untuk menjalankan sistem.

BAB IV

HASIL DAN PEMBAHASAN

2.15 Pengujian Hardware

Perancangan sistem *warning* sepeda motor berbasis SMS ini menggunakan perangkat yang seminimal mungkin agar alat tersebut dapat diterapkan atau dipasang pada sepeda motor tanpa banyak memakan tempat. Sistem *warning* sepeda motor yang dirakit terdiri atas AKI sepeda motor atau sumber tegangan 12 Volt DC, sebuah *handphone* sistem, unit relay, unit sensor *input*, dan sistem kendali atau unit pengolah. Skema Kerja Rangkaian sistem *warning* sepeda motor berbasis SMS dapat dilihat pada Gambar 4.1 berikut.



Gambar 4.1 Skema Kerja Rangkaian sistem *warning* sepeda motor berbasis

2.15.1 Pengujian *Handphone* Sistem

Pengujian *handphone* dilakukan dengan cara:

G. Menghubungkan *handphone* ke *port* serial PC menggunakan kabel data serial.

H. Membuka fasilitas *hyper terminal* pada *windows*, pilih *port* com yang terhubung dengan *handphone* dan atur setingan *port*-nya sesuai dengan tipe *handphone* yang digunakan.

I. Mengetikkan perintah ATE1 dan AT+CMGF=1 pada lembar kerja (pastikan *port* serialnya telah terkoneksi).

Berikut adalah hasil pengujian terhadap *handphone* siemens C35 yang dihubungkan dengan komputer menggunakan kabel data serial dengan memberikan perintah AT (AT-command).

Tabel 4.1 Hasil pengujian *handphone* sistem

PERINTAH	JAWABAN	KETERANGAN
ATE1	OK	HANDPHONE Mendukung Fasilitas AT-COMMAND
AT+CMGF=1	OK	HANDPHONE Mendukung Fasilitas AT-COMMAND dan Menggunakan Protokol Data Jenis PDU

Dari pengujian dengan mengetikkan perintah ATE1 dan AT+CMGF=1 yang telah dilakukan didapatkan jawaban OK, berarti *handphone* tersebut dapat dikontrol menggunakan perintah AT (AT-COMMAND) dan sistem komunikasinya menggunakan format PDU (*Protokol Data Unit*), sehingga *handphone* dapat digunakan sebagai *handphone* sistem yang terhubung dengan sistem mikrokontroler. Akan tetapi apabila jawabannya *error*, maka *handphone* tersebut tidak dapat digunakan sebagai *handphone* sistem.

2.15.2 Pengujian Mikrokontroler

Pengujian mikrokontroler dilakukan dengan cara :

- f. Memberikan logika '1' atau '0' ke kaki-kaki mikrokontroler yang dijadikan sebagai *port output*.
- g. Menghubungkan kaki-kaki mikrokontroler yang dijadikan *port output* dengan dengan led yang terhubung dengan VCC.

- h. Menghubungkan pin 40 IC mikrokontroler dengan Vcc sebesar 5 volt dan pin 20-nya dihubungkan dengan ground.
- i. Pin *reset* yang berfungsi berfungsi untuk memulai perintah dari alamat pertama, dihubungkan dengan Vcc yang diberi kapasitor dan resistor untuk menstabilkan tegangan yang masuk dari Vcc. Selain itu untuk melakukan proses mengolah data mikrokontroler membutuhkan suatu clock yang berperan untuk menentukan kecepatan kerja. Rangkaian pembangkit pulsa terdiri dari *crystal* dan dua buah kapasitor yang dihubungkan dengan pin XTAL1 dan XTAL2.

Pengujian mikrokontroler dilakukan dengan melakukan pengetesan setiap *port* dengan memasukkan program sederhana. Misal dengan memasukkan program untuk menyalakan led. Program yang digunakan seperti pada Gambar 4.2. berikut :

```
$mod51

org    0000h

mov    p0,#00000011b    ;port0

mov    p1,#00001100b    ;port1

mov    p2,#00110000b    ;port2

mov    p3,# 11000000b    ;port3

end
```

Gambar 4.2. program untuk menyalakan LED

Dari program di atas diperoleh hasil tampilan led seperti pada Tabel 4.2 berikut :

Tabel 4.2 Tampilan LED

No.	Port	Tampilan led
1.	0	○ ○ ○ ○ ○ ○ ● ●
2.	I	○ ○ ○ ○ ● ● ○ ○
3.	II	○ ○ ● ● ○ ○ ○ ○
4.	III	● ● ○ ○ ○ ○ ○ ○

Dari hasil pengujian yang didapat, menunjukkan bahwa program dapat berjalan dengan baik, sehingga mikrokontroller dapat digunakan sebagai komponen pengendali sistem.

2.15.3 Interaksi *handphone* dengan sistem mikrokontroller

Komunikasi antara sistem mikrokontroler dengan *handphone* sistem menggunakan kabel data serial dengan standar RS-232. Untuk mengubah level tegangan TTL mikrokontroler menjadi level tegangan RS-232 *handphone* digunakan IC MAX-232. Untuk menjalankan sistem *warning* pada sepeda motor tahap pertama yang diproses adalah penginisialisasian *port* serial yang merupakan jalur penghubung antara *handphone* dengan rangkaian mikrokontroller, apabila sistem mikrokontroller tidak mendeteksi adanya *handphone* yang terhubung dengan *port* serial maka sistem mikrokontroller belum melakukan pengaktifan sistem *warning*.

Pengujian intraksi *port* serial antara *handphone* dengan mikrokontroller ini dilakukan dalam beberapa langkah sebagai berikut:

- 2) Rangkaian Mikrokontroler dihubungkan dengan dengan power supply 12 V yang nantinya akan dikonversi menjadi 5 Volt oleh IC 7805.
- 3) IC Mikrokontroller terlebih dahulu harus diprogram untuk dapat melakukan penginisialisasian *port* serial.
- 4) *Port* serial yang terhubung dengan sistem mikrokontroller dihubungkan dengan kabel data serial yang terhubung dengan *handphone* sistem (*Handphone* dalam keadaan aktif).

Penginisialisasian *port* serial dikatakan berhasil apabila *port* 3.1 pada mikrokontroller setelah dihubungkan dengan *handphone* sistem mengalami perubahan tegangan (berlogika '0' / aktif *low*). Dari program yang telah dibuat, apabila sistem dapat melakukan komunikasi dengan *handphone* maka *port* P3.7 akan aktif (berlogika '0' / aktif *low*), sebaliknya apabila sistem tidak dapat terkoneksi dengan *handphone* maka P3.7 dalam keadaan tidak aktif.

Apabila proses penginisialisasian *port* serial telah berhasil kemudian dilakukan pengecekan terhadap perintah SMS yang diterima dari *handphone* sistem. Sistem mikrokontroller akan mengolah *input*-an SMS yang berasal dari dan hanya dari nomor *handphone* yang telah ditetapkan dalam program. Sistem akan selalu dalam keadaan aktif ketika sistem dihubungkan dengan catu daya. Sistem hanya dapat mengolah perintah SMS yang berasal dari *handphone* user. Apabila ada penerimaan perintah SMS ke *user* sistem yang tidak berasal dari nomor *user* yang ditetapkan dalam program (*user*), maka perintah SMS tersebut akan diabaikan dan dihapus. Namun apabila terdapat penerimaan perintah SMS yang berasal dari nomor *user* yang telah ditetapkan dalam

program maka perintah tersebut akan diolah oleh sistem mikrokontroller.

2.15.4 Pemicu Relay

Sistem *warning* sepeda motor berbasis SMS ini menggunakan unit relay yang berfungsi sebagai saklar otomatis untuk memutus atau menghubungkan jalur- jalur kelistrikan pada sepeda motor yang terhubung dengan AKI. Relay juga digunakan untuk memberikan catu daya ke komponen-komponen yang ada pada sepeda motor (klakson, *sign*, & starter). Apabila sistem *warning* diaktifkan maka sistem akan mendeteksi *input*-an yang diperoleh dari sensor-sensor yang terdapat pada kendaraan atau *input*-tan yang berupa perintah SMS yang berasal dari *user*, kemudian *input*-an tersebut akan diproses oleh sistem mikrokontroller yang selanjutnya akan dijadikan pemicu untuk menggerakkan masing-masing relay. Pengujian unit relay dilakukan untuk mengetahui apakah relay yang telah terhubung dengan driver relay (IC ULN 2803) bekerja sesuai dengan *input*-an yang diberikan.

Peralatan yang digunakan :

- a) Relay 12 volt
- b) Rangkaian driver pemicu relay (IC ULN 2803) yang terintegrasi dengan rangkaian mikrokontroller.
- c) Power Supply 12 Volt
- d) Multitester

Pengujian dilakukan dengan cara memberikan logika ‘1’ ke *port* mikrokontroler yang terhubung dengan *Port output* dari IC ULN 2803, kemudian *port* keluaran / *output* dari IC ULN 2803 diihubungkan dengan relay yang akan ditest, *port* com dari IC ULN 2803 dihubungkan dengan catu daya 12 volt dan *port* ground-nya dihubungkan dengan ground dari catu daya. Cara mengetahui relay apakah dapat bekerja atau tidak dengan cara menghubungkan kaki positif untuk kumparan relay dengan catu daya 12 volt, sedangkan kaki negatif dari kumparan relay dihubungkan dengan kaki *output* dari IC ULN 2803. Kaki-kaki dari relay yang terhubung dalam posisi *normali close* kemudian dihubungkan dengan multitester. Apabila pada saat relay diberikan

logika ‘1’ (4,8 - 5 volt) indikator atau jarum multimeter tidak menyimpang, berarti relay tidak dapat berfungsi dengan baik/ rusak. Akan tetapi jika dihubungkan pada posisi *normali open* mengakibatkan indikator atau jarum multimeter dapat menyimpang, berarti menunjukkan bahwa relay berfungsi dengan baik. Sebelum melakukan pengujian, seluruh rangkaian harus terhubung dengan sumber daya yang tepat agar tidak menyebabkan kerusakan pada komponen. Hasil pengujian terhadap *output* dari pemicu relay dapat dilihat pada Tabel 4.3 berikut.

Tabel 4.3. Hasil pengujian *output* pemicu relay

RELAY	Port Output Mikrokontroller	
	Logika 1	Logika 0
RELAY KONTAK / UTAMA	Aktif	Non aktif
RELAY STARTER	Aktif	Non aktif
RELAY BEL	Aktif	Non aktif
RELAY SIGN	Aktif	Non aktif

2.15.5 Komponen Sensor *Input*

Komponen sensor *input* merupakan rangkaian yang akan digunakan untuk memberikan logika ‘0’ atau ‘1’ pada *port input* mikrokontroller. Rangkaian sensor *input* terdiri dari tiga jenis yang masing-masing sensor memiliki perbedaan dalam pemasangannya, namun setiap sensor tersebut masing masing dihubungkan dengan optocoupler. Optocoupler berfungsi sebagai isolasi antara tegangan tinggi dengan tegangan rendah, dan juga berfungsi untuk menjaga tegangan yang berasal dari sensor menuju mikrokontroller agar tidak terjadi interferensi akibat penarikan kabel yang panjang.

Sensor yang pertama adalah sensor *reed switch*. Pengujian sensor ini dilakukan dengan cara menghubungkan salah satu kutub dari sensor *reed switch* dengan catu daya 12 volt, dan untuk ujung kutub yang satunya dihubungkan dengan optocoupler tapi harus diberikan hambatan sebesar 470 ohm. Sedangkan untuk kaki *collector* optocoupler dihubungkan dengan *port input* dari mikrokontroller, dan kaki *emitter*-nya dihubungkan dengan *ground*. Sensor ini dapat bekerja

apabila didekatkan dengan medan magnet, sehingga apabila sensor *reed switch* didekatkan dengan magnet maka *switch* akan terhubung sehingga akan mengalirkan tegangan ke optocoupler yang selanjutnya tegangan tersebut digunakan untuk memicu led yang terdapat pada optocoupler sehingga kaki *collector* dapat memberikan tegangan menuju *port input* dari mikrokontroller. Tegangan yang berasal dari kaki *collector* tersebut akan memberikan logika 0 pada *port input* dari mikrokontroller, sehingga logika tersebut akan diolah oleh sistem mikrokontroller untuk dijadikan pemicu dari unit relay dan juga untuk pemicu pengiriman SMS *warning*.

Sensor yang kedua adalah sensor posisi kunci kontak. Sensor ini hanya terdiri dari optocoupler, hambatan, dan *switch*. *Switch* yang digunakan dari rangkaian sensor adalah *switch* dari kontak kendaraan, dimana apabila posisi kontak dalam keadaan ON maka *switch* akan terhubung, sehingga tegangan dapat mengalir ke optocoupler untuk memicu led. Akan tetapi tegangan yang masuk ke optocoupler terlebih dahulu diberikan hambatan 470 ohm untuk mengurangi arus yang lewat ke optocoupler. Pengujian sensor ini hampir sama dengan pengujian untuk sensor *reed switch*, dimana apabila kaki-kaki *input* dari optocoupler dihubungkan dengan catu daya 12 volt maka akan menyebabkan led yang ada dalam optocoupler tersebut aktif, kemudian akan memicu tegangan pada kaki *collector* yang dihubungkan dengan *port input* dari mikrokontroller. Tegangan tersebut akan menyebabkan kaki *input* dari mikrokontroller menjadi berlogika 1, kemudian inputan tersebut akan dijadikan pemicu untuk mengaktifkan atau menonaktifkan relay.

Sensor ketiga adalah sensor indikator gigi normal/ *neutral*. Sistem kerja dari sensor ini hampir sama dengan kedua sensor sebelumnya, yaitu dengan mempergunakan pemicu untuk mengaktifkan led pada optocoupler, kemudian apabila led telah aktif berarti optocoupler akan memberikan tegangan ke pin *input* dari mikrokontroller (logika '1'), kemudian *output*-an dari sensor tersebut akan diolah oleh sistem mikrokontroller. Cara pemasangan sensor ini yaitu kaki led anoda optocoupler dihubungkan secara paralel dengan kutub positif dari lampu indikator gigi normal/ *neutral*, dalam rangkaian yang telah dibuat diberikan hambatan sebesar 470 ohm yang berfungsi sebagai penghambat arus dari tegangan 12 volt ke kaki led anoda optocoupler. Sedangkan untuk kaki led katoda optocoupler dihubungkan dengan kutub negatif dari lampu indikator gigi normal yang terhubung dengan *ground*. Sehingga apabila lampu indikator gigi normal menyala, berarti kutub positif dari lampu tersebut mendapat tegangan 12 volt dan tegangan tersebut akan masuk ke optocoupler yang dijadikan pemicu untuk mengaktifkan led, kemudian

apabila led optocoupler aktif maka optocoupler akan memberikan tegangan melalui kaki *collector* optocoupler menuju *port input* dari mikrokontroller, sehingga menyebabkan kaki *input* mikrokontroller mendapat logika ‘1’. Inputan dari optocoupler tersebut akan diproses oleh mikrokontroler yang selanjutnya akan dijadikan data sebagai pemicu relay starter dari jarak jauh (melalui SMS). Hasil pengujian dari sensor input dapat dilihat pada Tabel 4.4. berikut.

Tabel 4.4. Hasil pengujian sensor *input*

SENSOR	Port input mikrokontroller	
	Logika 1	Logika 0
KEMUDI / REED SWITCH	Aktif	Non aktif
KONTAK	Aktif	Non aktif
GIGI NETRAL	Aktif	Non aktif

2.16 Pengujian Software

Sistem *warning* sepeda motor ini memiliki program yang digunakan untuk menjalankan keseluruhan sistem, diantaranya yaitu pengaktifan sistem pengaman (menonaktifkan relay utama/ klakson), pengaktifan sistem alarm (mengaktifkan relay klakson dan relay *sign*), serta untuk menjalankan proses pengiriman dan pengolahan data dari dan ke *user* sistem untuk proses pengiriman dan penerimaan perintah SMS.

Sistem alarm akan aktif apabila sistem mendapat SMS berupa perintah *input*-an yang dikirim dari nomor *user* yang telah ditetapkan dalam program. *Inputan* tersebut akan dijadikan pemicu untuk memberikan logika ‘1’ ke *port* keluaran dari mikrokontroller yang terhubung ke relay utama, sehingga apabila sistem diaktifkan maka relay utama akan terputus yang mengakibatkan terputusnya tegangan yang menuju ke starter, klakson dan *sign*. Sistem dalam keadaan aktif akan mendeteksi *inputan* yang berasal dari sensor *reed switch*, sensor posisi kunci kontak, dan interupsi dari *port* serial (perintah SMS).

Apabila sensor *reed switch* terdeteksi dan sistem alarm dalam keadaan aktif berarti *port*0.1 mendapat logika ‘1’ yang akan menyebabkan *port*2.2 dan *port*2.3 dalam keadaan high selama 2 menit (relay utama masih dalam keadaan terputus). Sistem mikrokontroler juga akan melakukan pengiriman SMS *warning* (“Sensor Kemudi ON”) ke nomor *user* yang telah ditetapkan dalam program (6285647198219).

Apabila sensor posisi kontak ON terdeteksi dan sistem alarm dalam keadaan aktif berarti

port0.0 mendapat logika ‘1’ yang akan menyebabkan *port2.2* dan *port2.3* dalam keadaan high selama 2 menit (relay utama masih dalam keadaan terputus). Sistem mikrokontroler juga akan melakukan pengiriman SMS *warning* (“sensor kontak ON”) ke nomor *user* yang telah ditetapkan dalam program (6285647198219).

Apabila sistem mendapat interupsi dari *port* serial (mendapat perintah SMS) yang berasal dari nomor *user* yang telah ditetapkan dalam program (6285647198219), maka interupsi tersebut akan dijalankan dan selanjutnya sistem mikrokontroler akan mengecek data yang diterima dari *port* serial tersebut (berupa data PDU) untuk kemudian bit data perintahnya diproses untuk dijadikan pemicu untuk mengaktifkan relay. Listing program dari sistem *warning* sepeda motor ini terdapat dalam Lampiran 3 halaman 71. Hasil pengujian instruksi SMS untuk mengaktifkan relay dalam keadaan sensor ON beserta keterangannya dapat dilihat pada Tabel 4.5. berikut.

Tabel 4.5. Hasil pengujian sistem ketika diberi perintah SMS

PERINTAH SMS	KEADAAN RELAY				KERANGAN
	RELAY UTAMA	RELAY BEL	RELAY SIGN	RELAY STARTER	
0	ON	OFF	OFF	OFF	ALARM OFF
1	OFF	ON	ON	OFF	ALARM ON
2	OFF	ON	ON	OFF	SIGN NYALA
3	OFF	OFF	OFF	OFF	JALUR KELISTRIKAN TERPUTUS
4	OFF	OFF	OFF	ON	STARTER ON

Apabila sistem mendeteksi adanya pengaktifan sensor, baik sensor kemudi(*reed switch*) ataupun sensor dan sistem alarm dalam keadaan aktif, selain sistem akan menyalakan relay klakson dan relay *sign*, sistem juga akan mengirimkan SMS peringatan kepada *user*. Hasil pengujian dari sistem peringatan melalui perintah SMS dapat dilihat pada Tabel 4.6. berikut.

Tabel 4.6. Hasil pengujian sistem peringatan

Sensor aktif	Kondisi sistem	Port aktif	SMS peringatan
Kemudi / <i>reed switch</i>	Aktif	P0.1	Sensor Kemudi ON
Kontak	Aktif	P0.0	Sensor Kontak ON

Hasil pengujian terhadap sistem *warning* sepeda motor jarak jauh berbasis SMS ini dilakukan pada saat kondisi operator jaringan yang digunakan dalam keadaan normal.

Hasil jadi dari rancang bangun sistem *warning* sepeda motor berbasis mikrokontroler

dapat dilihat pada lampiran 4 halaman 81.

BAB V

PENUTUP

5.5 Kesimpulan

Dari keseluruhan proses pembuatan tugas akhir yang telah dilakukan, dapat diambil kesimpulan sebagai berikut :

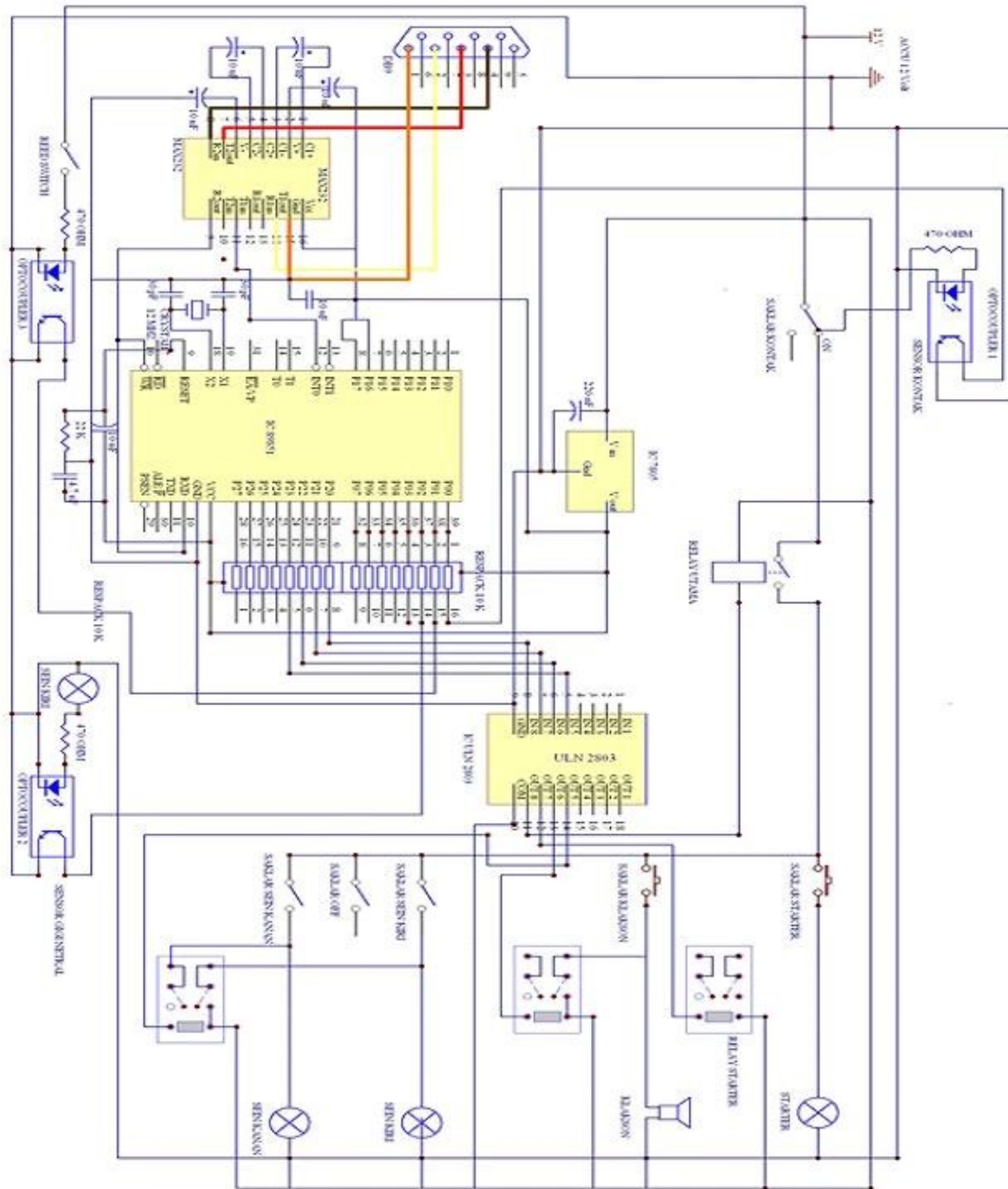
- 3 Berhasil melakukan rancang bangun sistem pengaman sepeda motor jarak jauh dengan memanfaatkan teknologi SMS (*Short Message Service*) serta menggunakan mikrokontroler AT89S51.
- 4 Prinsip utama sistem *warning* sepeda motor berbasis SMS yang telah dibuat ini bekerja dengan cara melakukan pengaktifan atau penonaktifan unit relay yang dihubungkan dengan komponen-komponen kelistrikan yang ada pada sepeda motor.
- 5 Semua percobaan alat, dilakukan pada sebuah media simulator.

5.6 Saran

Setelah terselesaikannya tugas akhir dan mendapatkan hasil uji kerja dari sistem *warning* sepeda motor jarak jauh berbasis SMS ini, saran dari penulis antara lain :

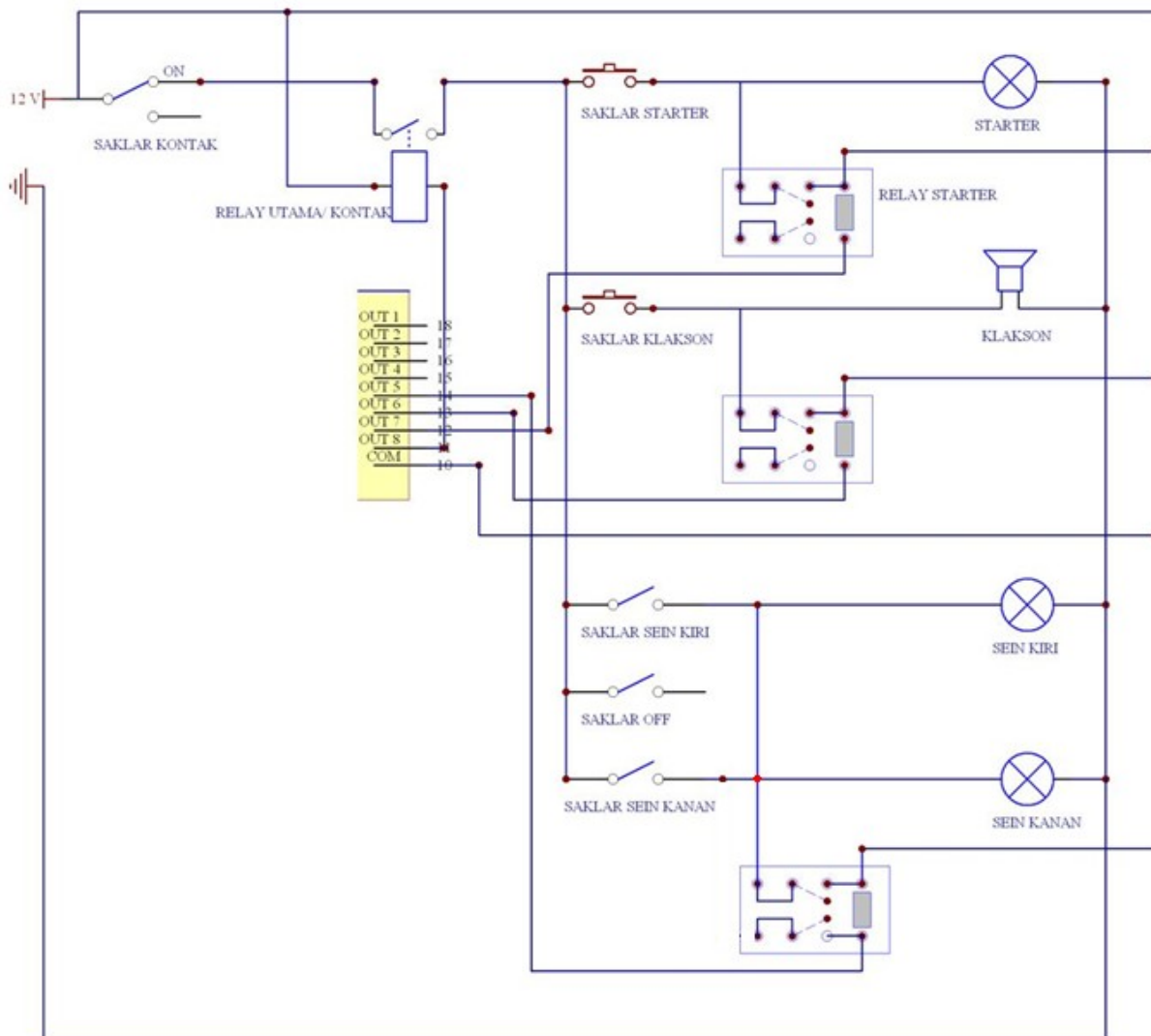
26. Sistem kendali jarak jauh dari sistem ditambah dengan menggunakan *misscall*.
27. Ditambahkan sensor mesin yang berguna untuk menghidupkan mesin melalui jarak jauh, sehingga apabila mesin telah hidup maka sistem dapat memberikan pemberitahuan kepada pemilik kendaraan.
28. Sistem dilengkapi catu daya tambahan apabila catu daya dari aki tidak berfungsi.
29. Menambahkan rangkaian untuk menyuplai daya ke handphone sistem, sehingga tidak perlu khawatir apabila handphone kehabisan daya.

LAMPIRAN 1 Rangkaian Sistem Mikrokontroler



LAMPIRAN 2

Rangkaian Unit Keluaran dan Relay



LAMPIRAN 3

Soure Code Program Keseluruhan Sistem warning SMS

```
=====
$mod51
skontak      bit    p0.0      ;aktif low
sreed        bit    p0.1
snetral      bit    p0.2
smesin       bit    p0.3

relay        equ    p2
rkontak      bit    p2.0      ;aktif  high
rstart       bit    p2.1
rbel         bit    p2.2
rsign        bit    p2.3

dtr          bit    p3.2
falarm       bit    p3.5
connect      bit    p3.7

flag:        dseg    at 20h
             ds      1
o_pdu:       ds      1
dthp:        ds      90

             cseg    at 0h
             jmp     start

             org     0bh      ;timer 0 interupt
             jmp     timer0

             org     23h      ;serial interupt
             jmp     serin

start:       mov     relay,#0
             mov     tmod,#21h ;t1 8 bit autoreload,
             mov     th1,#0fdh ;9600 BPS
             orl     pcon,#10000000b ;SMOD=1 --> 19200 BPS
             setb    tr1      ;timer 1 run
             mov     scon,#50h ;serial mode 1
             mov     ip,#02   ;priority timer 0 high
             setb    ea       ;enable all interupt
             setb    es       ;enable serial
```

```

setb    et0            ;enable timer 0
clr     dtr            ;ignition on
call    clr_ram
er_con: call    delay
mov     dptr,#p_ate0   ;kirim perintah
call    command        ;AT
call    delay
mov     th0,#0
setb    tr0
jnb     f0,$           ;tunggu jawaban
clr     f0
mov     dptr,#j_ate0   ;bandingkan
call    cekking        ;jawaban
mov     a,flag
cjne    a,#1,er_con    ;salah,--> ulang
clr     connect        ;benar,--> led connect on
mov     dptr,#p_cnmi
call    command
call    delay
jnb     f0,$           ;tunggu jwb cnmi
clr     tr0
clr     f0
; jmp     csms          ;*****
lagi:   jb     skontak,creed ;sensor kontak
jnb     falarm,alarmof
clr     rkontak        ;kontak off
setb    rbel           ;bel on
setb    rsign          ;sign on
mov     dptr,#s0000
call    sendsms
call    dalarm
clr     rbel
clr     rsign
jmp     creed
alarmof: setb    rkontak        ;kontak on
creed:  jb     sreed,csms      ;sensor reed sw
jnb     falarm,csms      ;sistem off, cek sms
setb    rbel           ;bel on
setb    rsign          ;sign on
mov     dptr,#s0001      ;kirim sms alarm
call    sendsms
call    dalarm
clr     rbel
clr     rsign
csms:   jnb     f0,lagi        ;tunggu sms *****
clr     tr0

```

```

clr    f0
call   delay
mov    dptr,#p_read ;kirim perintah
call   command      ;baca sms
call   delay
jnb    f0,$          ;tunggu jawaban
clr    tr0
clr    f0
call   proses        ;proses data
call   clr_ram
mov    dptr,#p_del1 ;kirim perintah
call   command      ;hapus sms
call   delay
jnb    f0,$          ;tunggu jawaban
clr    tr0
clr    f0

mov    dptr,#p_del2 ;kirim perintah
call   command      ;hapus sms
call   delay
jnb    f0,$          ;tunggu jawaban
clr    tr0
clr    f0

mov    dptr,#p_del3 ;kirim perintah
call   command      ;hapus sms
call   delay
jnb    f0,$          ;tunggu jawaban
clr    tr0
clr    f0

mov    dptr,#p_del4 ;kirim perintah
call   command      ;hapus sms
call   delay
jnb    f0,$          ;tunggu jawaban
clr    tr0
clr    f0
JMP    csms          ;lagi

```

```

;=====

```

```

;Clear Ram

```

```

;-----

```

```

clr_ram:  mov    r1,#dthp
           mov    b,#90
clear:    mov    @r1,#255 ;clear dthp
           inc    r1

```

```

    djnz    b,clear
    mov     dthp,#254
    ret

```

```

;=====

```

```

;Kirim perintah

```

```

;-----

```

```

command:    clr     f0
            mov     r1,#dthp
cmd:         mov     a,#0
            movc    a,@a+dptr
            cjne    a,#255,lj      ;255 akhir data
            ret                     ;selesai
lj:          clr     ti             ;
            mov     sbuf,a          ;kirim perintah
            jnb     ti,$
            inc     dptr            ;perintah selanjutnya
            jmp     cmd

```

```

;=====

```

```

;Interupsi Serial disimpan pada alamat r1

```

```

;-----

```

```

serin:      jnb     ri,xsr          ;rx?
            setb    tr0             ;ya--> matikan timer
            mov     r1,#dthp
slg:        mov     th0,#-30
            mov     @r1,sbuf        ;simpan
            clr     ri              ;rx ready
            inc     r1              ;alamat berikutnya
lsr:        jb      ri,slg
            jb      tr0,lsr
xsr:        reti

```

```

;=====

```

```

;Serial time out

```

```

;-----

```

```

timer0:     clr     tr0             ;timer off
            MOV     TH0,#-30
            setb    f0
            mov     @r1,#255
XTIM0:      reti

```

```

;=====

```

```

;Cek Jawaban HP; dptr=jml,referensi; sama --> flag=1

```

```

;-----

```

```

ceking:      mov    flag,#1      ;flag benar
             mov    r1,#dthp
u_cek:       mov    a,#0
             movc   a,@a+dptr    ;jml referensi
             cjne   a,#255,l_cek
             ret
l_cek:       mov    b,@r1        ;ambil jwb hp
             cjne   a,b,ljtx     ;bandingkan
             inc    dptr         ;sama lanjutkan
             inc    r1
             jmp    u_cek        ;semua sama, --> flag=benar
ljtx:        mov    flag,#0      ;tidak sama, flag=salah --> selesai
             ret

```

```

;=====

```

```

;PDU; r1=data hp

```

```

;-----

```

```

proses:      mov    r1,#dthp+38  ;36 offset nomor pengirim
             mov    dptr,#no_hp  ;bandingkan

```

```

pdu2:        mov    b,@r1
             ;call   cektx
             mov    a,#0
             movc   a,@a+dptr
             cjne   a,#255,pdu1
             jmp    pdu3         ;sama? --> lanjut

```

```

pdu1:        ;mov    b,a          ;ignore no hp
             cjne   a,b,xpdu     ;tidak? --> hapus
             inc    r1
             inc    dptr
             jmp    pdu2

```

```

pdu3:        mov    r1,#dthp+72  ;72   offset kode on/off

```

```

;MOV A,@R1
;CALL CEKTX
;INC R1
;MOV A,@R1
;CALL CEKTX
;DEC R1

```

```

    call   hex2
    anl    a,#01111111b  ;ambil 7 bit

```

```

;call   cektx
    cjne   a,'#0',on1
    clr    falarm        ;alarmoff
    ;setb  rkontak
    ret

```

```

on1:         cjne   a,'#1',on2

```

```

        setb    falarm        ;alarm on
        clr     rkontak
        ret
on2:     cjne    a,'#2',on3    ;find on
        setb    rsign         ;sign on
        setb    rbel          ;bel on
        call    delay
        clr     rbel          ;bel off
        call    dalarm
        clr     rsign         ;sign off
        ret
on3:     cjne    a,'#3',on4
        clr     rkontak        ;mesin off
        ret
on4:     cjne    a,'#4',xpdu
        jb      snetral,nostart
        setb    rkontak
        call    dstart
        setb    rstart        ;starter
        call    dstart
        clr     rstart
xpdu:    ret
nostart: setb    rbel
        call    delay
        clr     rbel
        ret

;=====
;Kirim sms
;-----
sendsms: push    dph
        push    dpl
        mov     b,#10         ;tunda 1 detik
ccst:    call    delay
        djnz    b,ccst
        mov     dptr,#p_send  ;kirim AT+cmgr=28
n_send:  mov     a,#0
        movc    a,@a+dptra
        cjne    a,#255,lsend
        jmp     s_pdu
lsend:   call    cektx
        inc     dptr
        jmp     n_send
s_pdu:   call    delay
        jnb     f0,$          ;tunggu jawaban
        clr     f0

```

```

        clr    tr0
        mov    b,#10          ;tunda 1 detik
ls_pdu:  call    delay
        djnz   b,ls_pdu

        pop    dpl
        pop    dph
ntxpdu:  mov     a,#0
        movc   a,@a+dptr
        cjne   a,#255,txpdu
        call   delay
        jnb    f0,$           ;tunggu jawaban
        clr    f0
        clr    tr0
        ret
txpdu:   call   cektx         ;kirim pesan
        inc    dptr
        jmp    ntxpdu

```

```

;=====

```

```

;conversi ascii ke hexadecimal

```

```

;-----

```

```

hex2:    call   asc2hex       ;digit 2
        mov    b,#16
        mul    ab
        push   acc
        inc    r1
        call   asc2hex
        mov    b,a
        pop    acc
        add    a,b
        inc    r1
        ret
asc2hex:  mov    a,@r1        ;digit 1
        clr    c
        subb   a,#30h
        cjne   a,#10,asc1
asc1:    jc     asc2
        clr    c
        subb   a,#7
asc2:    ret

```

```

;=====

```

```

;Tx Acc

```

```

;-----

```

```

cectx:      clr    ti
            mov    sbuf,a
            jnb    ti,$
            CLR    ti
            ret

;=====
;Delay
;-----
delay:      mov    r7,#0
dly:        mov    r6,#0
            djnz   r6,$           ;256*2us=512uS
            djnz   r7,dly        ;256*512uS=131.072uS
            ret                ;0,13 S

Dalarm:      mov    r5,#200
dalarm1:    call   delay          ;0,13S
            djnz   r5,dalarm1    ;200*0,13=26S
            ret

Dstart:      mov    r5,#15
dstart1:    call   delay          ;0,13*15=1,95S
            djnz   r5,dstart1
            ret

;=====
;tabel
;-----
p_ate0:      db 'ATE0',13,10,255
p_cnmi:      db 'AT+CNMI=1,1,2,2,1',13,10,255
p_read:      db 'AT+CMGR=1',13,10,255
p_del1:      db 'AT+CMGD=1',13,10,255
p_del2:      db 'AT+CMGD=2',13,10,255
p_del3:      db 'AT+CMGD=3',13,10,255
p_del4:      db 'AT+CMGD=4',13,10,255
no_hp:      db '265846178912F9',255
j_ate0:      DB 13,10,'OK',255

end

```


LAMPIRAN 4

Foto Rangkaian Sistem Warning Sepeda Motor Jarak Jauh Berbasis SMS

